

平成 30 年度修士論文

二重 SQUID 構造を採用した単一磁束 量子デジタル信号増幅回路の研究

電気通信大学 大学院

基盤理工学専攻

電子工学プログラム

学籍番号 1733103

氏名 樋口 孔明

主任指導教員 水柿 義直 教授

指導教員 島田 宏 教授

提出日 平成 31 年 2 月 27 日

概要

単一磁束量子を情報担体とした単一磁束量子 (RSFQ: Rapid Single-Flux-Quantum) 回路は、その高速性と優れた低消費電力から次世代の回路としての期待がされており、研究が盛んに進められている。しかし、SFQ の持つエネルギーの低さから低温の SFQ 回路と室温の半導体回路を橋渡しするインターフェース回路が必要となる。以上の理由からオンチップ上の RSFQ 回路出力増幅回路の研究が進められている。その中の 1 つに分布型アンプ (Distributed Amplifier) がある。この回路は従来の増幅回路と伝送線路の技術を組み合わせることで広帯域幅と高利得の両立を可能にした回路であり、超伝導回路としての先行研究もされている。この先行研究では正常に動作を観測したが、出力振幅が小さいため室温環境との通信ではより高い振幅電圧を得る必要がある。そこで本研究はこのオンチップ RSFQ 分布型アンプの出力電圧向上に向けた改良を行い、その特性を評価することを研究目的とした。出力電圧の向上には段数の増加と 1 段あたりの出力振幅を増加させる 2 つのアプローチがあるが、我々は後者のアプローチを取った。後者のアプローチを取った理由としては、段数を増やすとその分遅延時間も増加してしまい結果として帯域幅が減少してしまうためである。以降先行研究の SQUID が 1 個の分布型アンプを単一 SQUID 構造分布型アンプ (DSSA: Distributed Single-SQUID Amplifier)、我々が設計した SQUID が 2 個の分布型アンプを二重 SQUID 構造分布型アンプ (DDSA: Distributed Double-SQUID Amplifier) と記載する。数値計算では 4 段、12 段、24 段二重 SQUID 構造分布型アンプを設計、単一 SQUID 構造分布型アンプとの比較を行なった。数値計算での結果では 4 段二重 SQUID 構造分布型アンプの最大出力振幅は 4 段単一 SQUID 構造の回路と比較して 2.07 倍に増加し、最大動作周波数は 9.5Gbps と求まった。12 段回路の最大出力振幅は 10.15mV、24 段回路では 19.55mV と求まり、単一 SQUID 構造の回路と比較して 2.00 倍、1.96 倍に増加した。数値計算で動作を確認したため、実際に設計、作製を行い、特性の評価を行った。4 段回路では低速測定において最大出力振幅 2.93mV を観測し、単一 SQUID 構造の回路での最大出力振幅 1.94mV と比較して 1.51 倍に増加した。数値計算よりも低い増加率となったが、これは二重 SQUID 構造に起因する。4 段二重 SQUID 構造分布型アンプの動作を確認したため、12 段と 24 段二重 SQUID 構造分布型アンプの低速測定も実施した。12 段二重 SQUID 構造分布型アンプの最大出力振幅は 8.23mV、24 段二重 SQUID 構造分布型アンプの最大出力振幅は 14.50mV であったが、数値計算と比較するとバイアス電流に対する動作領域と最大出力振幅のどちらも低いという結果になった。また 12 段二重 SQUID 構造分布型アンプで高速測定を実施したが、完全な動作とはならず部分的に動作していることが確認できるのみだった。低速測定では動作を確認できたため、パラメータの改良を進めていくことで高速測定においても動作実証が可能になると考えられる。

目次

第Ⅰ章	序論	5
1	研究背景	5
2	研究目的	5
第Ⅱ章	原理	7
3	超伝導の特性	7
3.1	完全導電性	7
3.2	マイスナー効果	8
3.3	磁束の量子化	10
3.4	ジョセフソン効果	11
4	dc-SQUID	13
5	単一磁束量子回路	16
6	ジョセフソン伝送線路	17
7	SFQ 論理セル	21
7.1	Splitter	21
7.2	Confluence Buffer	22
7.3	TFF	22
7.4	NDRO	23
7.5	dc/SFQ	24
7.6	SFQ/dc	25
8	超伝導出力増幅回路	27
8.1	RSFQ 分布型アンプ (RSFQ Distributed Amplifier)	27
8.2	RSFQ 分布型アンプの動作原理	29

第 III 章	回路設計及び試作方法	32
9	Nb 集積回路作製プロセスの設計仕様	32
10	SFQ 論理セルライブラリ (CONNECT Cell Library)	34
11	超伝導集積回路のシミュレーション	34
12	作製回路のレイアウト設計	35
12.1	低速測定	35
12.2	高速測定	38
第 IV 章	測定方法	40
13	出力振幅の測定方法 (低速測定)	40
14	出力振幅の測定方法 (高速測定)	40
第 V 章	測定結果及び考察	43
15	数値計算	43
15.1	最大出力振幅	43
15.2	バイアス電流に対する動作領域	48
15.3	入力周波数に対する出力振幅特性	49
16	低速測定結果	51
16.1	測定結果と計算結果の比較	51
16.2	二重 SQUID 構造分布型アンプと単一 SQUID 構造分布型アンプの測定結果の比較	58
17	高速測定結果	60
第 VI 章	結論	62
	謝辞	67
付録 A	単一 SQUID 構造分布型アンプの結果	69

A.1	設計した回路のパラメータ	69
A.2	測定結果	70
A.3	考察	72
付録 B	数値計算で用いた jsim ファイル	73
付録 C	数値計算で用いた補助プログラム	83

第 I 章 序論

1 研究背景

超伝導は 1911 年に Kamerlingh Onnes らによって水銀を冷却することで電気抵抗が消失したことから発見された [1]。この発見を皮切りに、超伝導現象の機構の解明やより高温での超伝導現象、超伝導現象を用いた応用研究など数多くの研究が現在までに行われてきた。超伝導体の性質は主に完全導電性、マイスナー効果、磁束の量子化、ジョセフソン効果の 4 つがあり、いずれも常伝導状態では見られない不思議な現象である。中でも超伝導体間に薄い絶縁層を挟んだトンネル接合はジョセフソン接合と呼ばれ、この接合によって生じるジョセフソン効果はその特性から超伝導エレクトロニクスへの応用研究が盛んに進められている。我々はその超伝導エレクトロニクスの中で、磁束を情報担体として扱う単一磁束量子 (SFQ: Single Flux Quantum) 回路の研究を行っている。この回路は超伝導ループ内を磁束が貫く際に、貫く磁束の数が整数倍の値を取る (磁束の量子化) という現象を利用した回路である。この量子化された磁束を磁束量子と呼び、ループ内を磁束が貫いているかを論理値の "1"、"0" に当てはめて演算を行う。この回路は 1970 年代に中島ら [2, 3] によって考案され、その後 Likharev らによって体系化された [4]。Likharev らによって体系化された最も基本的な回路方式が RSFQ (Rapid SFQ) 回路であり、我々はこの回路方式を使用して研究を進めている。この回路は回路動作に必要なバイアス電流と接合を磁束が通り抜ける (スイッチすると呼ぶ) 際に生じるエネルギーの他に消費電力がゼロかつスイッチング速度が ps オーダーと非常に高速なことから従来の半導体回路に変わる次世代の回路として研究が行われており、数 100GHz での動作実証も行われている [5]。しかし、SFQ 1 個のエネルギーが 2.07×10^{-15} Wb と非常に小さく、低温状態では問題は無いが室温回路で観測を行おうとすると熱雑音に紛れて観測できなくなってしまうという問題点がある。

2 研究目的

この問題点を解決するための方法の 1 つとして、RSFQ 回路上で出力信号の増幅を行う方法がある。これは RSFQ 出力増幅回路と呼ばれており、RSFQ 回路を用いて SFQ パルス室温でも観測可能なレベルまで増幅する回路である。この回路については今までに様々な種類の増幅回路が考案されてきた。その内の 1 つに分布型アンプ (DA: Distributed Amplifier) がある。この回路は元々真空管時代に考案された回路であり [6, 7]、現在ではマイクロ波導波管で使用されている [8]。回路は CMOS などの従来の増幅回路と伝送線路からなり、直列に複数段接続することで利得と帯域幅の両方を維持することが可能な増幅回路である。

この回路は超伝導回路でも先行研究がされており、12 段直列接続した分布型アンプで 10Gbps のデータレートで最大出力振幅が 1.75mV という結果が得られている [9]。しかし室温回路への出力増幅回路として使うには出力振幅が小さく、より高い出力振幅が必要になる。高い出力振幅を得るだけで良いのなら段数を増やすことで実現できるが、段数を増やせば増やすほど回路の帯域幅も減少してしまうため、RSFQ 回路の利点が活かせなくなってしまう。

そこで本研究では、この分布型アンプの出力電圧向上に向けた改良と作製した回路の特性を調査した。出力電圧の向上には 1 段あたりの出力電圧を増加させる方法を採用し、帯域幅を維持しつつ出力電圧振幅を向上可能な改良を目的とした。

第 II 章 原理

3 超伝導の特性

超伝導体の特性として、主に完全導電性、マイスナー効果、磁束の量子化、ジョセフソン効果の 4 つの性質が挙げられる。

3.1 完全導電性

完全導電性とは、超伝導において電気抵抗が完全に無くなることで電圧降下なしに電流が流れる性質のことを指す。この時、電圧降下無しに流れ続ける電流は永久電流と呼ばれている。この性質は以下の図に示す実験によって観測することが出来る。

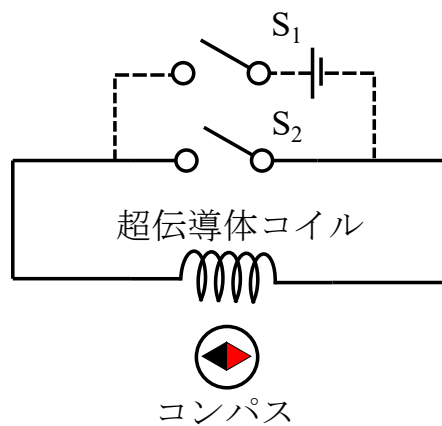


図 1 永久電流実験の概念図

実験の流れとしては、まず回路を超伝導体材料で作製して回路を転移温度よりも低い温度に冷却する。次にスイッチ S_1 を入れてリング上に電流を流してからスイッチ S_2 を入れてスイッチ S_1 を切る。本来ならこの時点で電流は流れなくなるが、コイルを含む超伝導リングではスイッチ S_1 を切っても電流が流れ続ける。これによってコイルに生じた磁界も消えることなく維持される。磁界が維持されていることはこの状態でコイル付近にコンパスを置くことによって確認することが出来る。

以上の実験によって超伝導状態では電気抵抗が 0 になることが分かったが、なぜ 0 になるのだろうか。常伝導状態ではキャリア、つまり電子の散乱によって電気抵抗が引き起こされる。その散乱は不純物原子や格子振動によって引き起こされ、格子振動は熱エネルギーに基づいて振動している。電気抵抗を無

くすにはこの2つの要因を取り除けばよく、不純物原子による散乱は純度を上げることで取り除くことが可能である。しかし格子振動に関しては0Kにおいてもゼロ点エネルギーによって格子振動は無くないため、取り除くことができない。そもそも電子が散乱するのは電子がフェルミ粒子として振る舞うためであるが、フェルミ粒子ではパウリの排他律によって1つのエネルギー準位にスピンの異なる2つの粒子しか存在できない。つまりエネルギー準位が最も低い状態では2つしか電子が入らないことになる。しかし、ボース粒子の場合はこの排他律は成り立たないので、1つのエネルギー準位に複数の粒子が存在できる。図2にボース粒子、フェルミ粒子のエネルギー状態の取り方を示す。

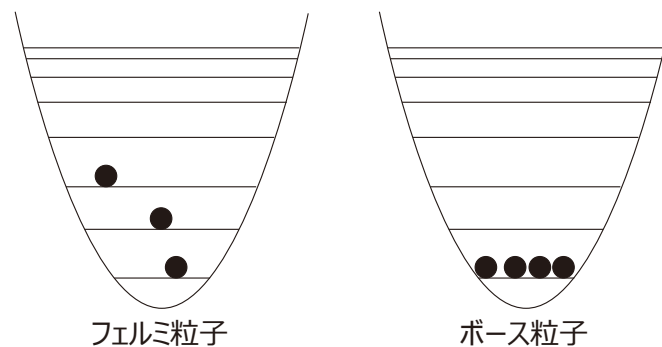


図2 フェルミ粒子とボース粒子のエネルギー状態

超伝導状態では電子が2つ対になって電子対となり、ボース粒子として振る舞うため最低準位においても複数の電子が存在できる。その結果格子振動による散乱を受けない状態となる。この電子対は超伝導電子対、クーパー対と呼ばれており、2つの電子はお互いの斥力ではなく格子振動(フォノン)による引力でペアとなっている。

3.2 マイスナー効果

超伝導状態では完全導電性によるゼロ抵抗だけでは説明できないマイスナー効果という現象がある。マイスナー効果は1933年にMeissnerとOchsenfeldによって発見され、完全反磁性と呼ばれる[10]。その名の通り、この現象は超伝導体内部の磁界を排除し、磁界が全く入らなくなるという現象である。

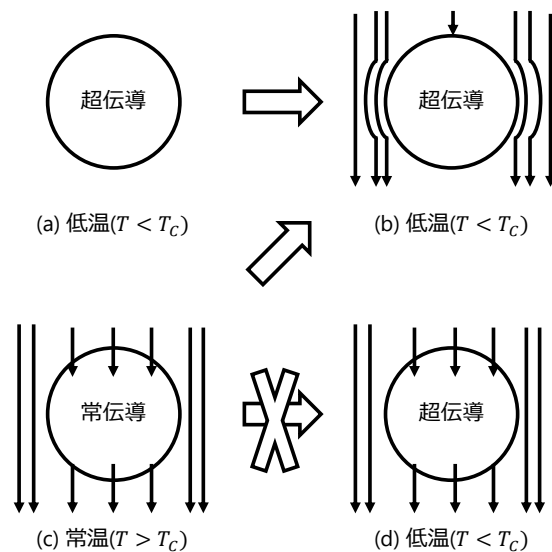


図3 マイスナー効果

上図はマイスナー効果について示した図である。まず図 (a) に示しているように超伝導体を臨界温度 T_C 以下の低温状態に保っておき、そこに図 (b) のように磁界 \mathbf{B} を加える。すると超伝導体の表面では磁界を打ち消す方向に電流が流れ、超伝導体内部には磁界は侵入しない。これはマクスウェル方程式から説明できる。超伝導状態では抵抗率 ρ が 0 であるので、電界 \mathbf{E} は

$$\mathbf{E} = \rho \mathbf{J} \quad (1)$$

と表され、電流が流れても磁界は発生しない。ここで \mathbf{J} は電流密度である。マクスウェル方程式

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E} \quad (2)$$

に代入するとこの値は 0 になり、超伝導体内部の磁束密度は変化しないということが分かる。

次に図 (c) のように常温の状態であらかじめ磁界を加えている場合を考える。この状態で超伝導の臨界温度 T_C 以下に下げた場合、完全導電性のみで考えると式 (2) によって超伝導内部の磁界の時間変化は 0 なので図 (d) のように内部には磁界が残るはずである。しかし、実際には図 (d) のような結果にはならず、図 (b) に示す通り超伝導体内部の磁界は排除され、内部には磁界は入らない。

この現象の説明は London 兄弟によって示された 2 つのロンドン方程式によって説明できる。以下に

2つのロンドン方程式を示す。

$$\frac{\partial}{\partial t} \mathbf{J}_S = \frac{nq^2}{m} \mathbf{E} \quad (3)$$

$$\nabla \times \mathbf{J}_S = -\frac{nq^2}{m} \mathbf{B} \quad (4)$$

式 (3) と式 (4) はそれぞれロンドンの第一方程式、第二方程式と呼ばれている。

ここで深さ方向が z の半無限長の超伝導体に外部から深さ方向に磁場 B_0 を加えた場合、ロンドン方程式を解くと、

$$B(z) = B_0 e^{-\frac{z}{\lambda_L}} \quad (5)$$

が得られる。この式は超伝導体の表面からの距離 z の関数として表されており、 λ_L はロンドン侵入長と呼ばれる磁束密度の侵入距離を表わす変数である。ロンドン侵入長は

$$\lambda_L = \sqrt{\frac{m}{\mu_0 n q^2}} \quad (6)$$

で与えられ、この2式より超伝導体に磁界を加えても超伝導体のごく表面のみまでしか磁束密度が侵入しないことが示される。

3.3 磁束の量子化

磁束の量子化とは、超伝導リング内を貫く磁束が、外部磁界の印加強度の強さに比例して飛び飛びの整数値をとる現象のことである。下の図4に示すのは磁束がリング内を貫く際に離散的に増加する様子である。

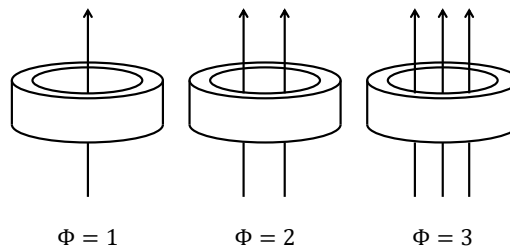


図4 磁束の離散的増加

この飛び飛びになっている磁束を磁束量子 (flux quantum) と呼び、その中でも 1 つだけ入っている状態の磁束を単一磁束量子 (SFQ:Single Flux Quantum) と呼ぶ。この磁束量子 1 つは以下に示す一定の値をとる。

$$\Phi_0 = \frac{h}{2e} = 2.07 \times 10^{-15} [\text{Wb}] \quad (7)$$

ここで、 e は素電荷 ($e = 1.602 \times 10^{-19} \text{C}$)、 h はプランク定数 ($h = 6.626 \times 10^{-34} \text{ m}^2\text{kg/s}$) である。

この現象が表れることは、超伝導体では量子力学的現象が巨視的に表れていることの証拠となる。磁束が Φ_0 の整数倍を取るということは、リングを流れる超伝導電流も整数倍になる、つまり波動関数として示せることを意味する。

3.4 ジョセフソン効果

ジョセフソン効果とはジョセフソン接合における電気的特性のことを指す。ジョセフソン接合は下の図のように、2 つの超伝導体の間に非常に薄い絶縁層を挟んだ接合のことをいう。

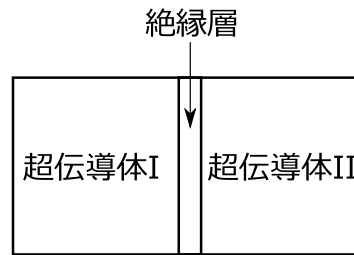


図 5 ジョセフソン接合図

クーパー対は電子と同様に波動性を持ち合わせており、1 つの超伝導体のクーパー対の位相は全て同じになる。しかし、絶縁層を挟んだ 2 つの超伝導体ではその位相が異なるために位相差 θ が生じる。生じた位相差 θ によってジョセフソン接合に電流が流れる。これが直流ジョセフソン効果である。この時の接合に流れる電流は、

$$I = I_C \sin \theta \quad (8)$$

上式で、 I_C は接合の臨界電流値であり、これはトンネル中を流れる最大のトンネル電流値である。臨界電流値を超える電流が流れると、接合間には電子対が崩壊して生じた電子が流れるようになり、特性は線形的になる。この時の I-V 特性を図 6 に示す。

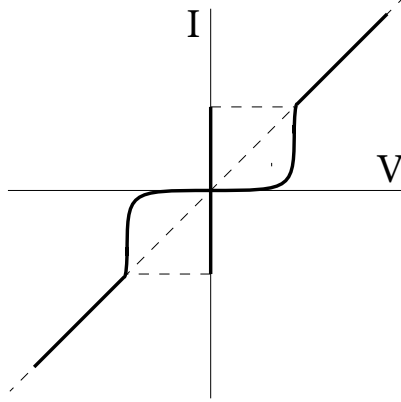


図6 ジョセフソン接合の IV 特性

この IV 特性では超伝導状態での特性が最初に生じ、その後常伝導状態の特性が表われる。一度常伝導状態になると、接合の臨界電流値以下に下げても電圧は出続ける状態になる。電流を止めることによって超伝導状態に戻るため、接合単体での振舞いはラッチ動作となる。

電流ではなく電圧が発生した場合、電圧によって位相差が変化し、位相は時間によって変化する。電圧が接合で発生した場合は交流電流が流れ、これが交流ジョセフソン効果となる。電圧が発生する場合の位相の時間変化は、

$$\frac{d\theta}{dt} = \frac{4\pi eV}{h} = \frac{2\pi V}{\Phi_0} \quad (9)$$

で示される。式 (9) を積分すると、

$$\theta = \frac{4\pi eV}{h}t + \theta_0 \quad (10)$$

となる。これを式 (8) に代入すると、

$$I = I_C \sin \left(\frac{4\pi eV}{h}t + \theta_0 \right) \quad (11)$$

となり、この時の交流電流の周波数 f は、

$$f = \frac{2eV}{h} \quad (12)$$

となる。

また、接合に周波数 f のマイクロ波を印加すると、電流-電圧特性に下図のような電流ステップが生じる。

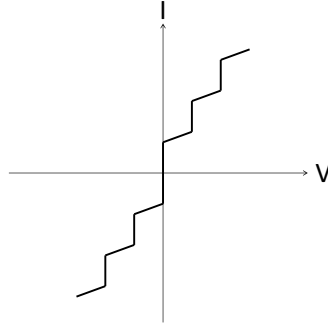


図7 シャピロステップ

このようなステップはシャピロステップと呼ばれており、シャピロステップの電圧 V は以下の式で示される。

$$V = \frac{h}{2e}nf = \Phi_0nf \quad (13)$$

n は自然数、 Φ_0 は物理定数であり磁束量子の値である。 Φ_0 は定数であるので、発生する電圧の精度は周波数 f にのみ依存する。この効果は、現在日本での直流電圧標準として採用されており、直流ジョセフソン効果に基づき 10^{-9} の精度の電圧を発生させている [11]。

4 dc-SQUID

SQUID は Superconducting QUantum Interference Device の略であり、超伝導量子干渉計と呼ばれるジョセフソン接合を用いて極微小な磁場を検出する高感度センサーとして利用されるデバイスである。地磁気よりも小さい磁場を検出可能なため、医療分野 [12] などでの応用が行われている。本研究ではこの SQUID を $\Phi - V$ 変換、つまり磁束から電圧に変換する回路として使用している。SQUID は超伝導リングの中に 1 つジョセフソン接合を含む rf-SQUID と 2 つ含む dc-SQUID の 2 種類が存在するが、ここでは dc-SQUID の方について述べる。

dc-SQUID の図を以下に示す。

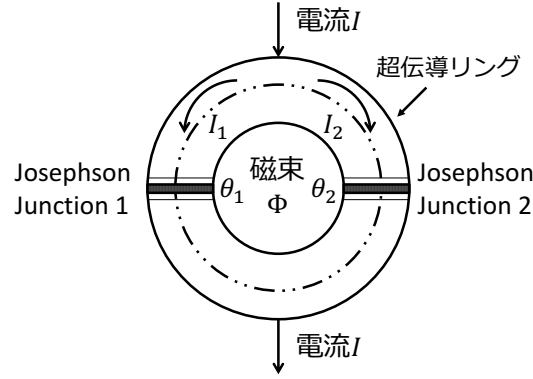


図 8 dc-SQUID

dc-SQUID は図 8 のように超伝導リング内に 2 つジョセフソン接合有する構造となっている。dc-SQUID 内の電流は接合の位相差を用いて以下の式で示される。

$$I_1 = I_{C1} \sin \theta_1 \quad (14)$$

$$I_2 = I_{C2} \sin \theta_2 \quad (15)$$

ここで、 I_{C1}, I_{C2} はそれぞれジョセフソン接合の臨界電流値である。

dc-SQUID 内の 2 つのジョセフソン接合が等しい特性であると仮定すると、外部磁界が印加されていない場合に両者を流れる電流は $I_1 = I_2, I_{C1} = I_{C2}$ という関係が成り立つ。従って、全体を流れる電流 I は、

$$I = I_1 + I_2 = 2I_{C1} \sin \theta_1 \quad (16)$$

となる。

一方外部磁界が印加された場合、外部磁界の影響を受けてジョセフソン接合 1 に流れる電流 I_1 とジョセフソン接合 2 に流れる電流 I_2 はそれぞれ変化する。よって全体を流れる電流 I は、

$$\begin{aligned} I &= I_1 + I_2 = I_{C1} \sin \theta_1 + I_{C2} \sin \theta_2 = I_{C1} (\sin \theta_1 + \sin \theta_2) \\ &= 2I_{C1} \sin \frac{\theta_1 + \theta_2}{2} \cos \frac{\theta_1 - \theta_2}{2} \end{aligned} \quad (17)$$

となる。この時の I の最大値を I_{MAX} とすると、

$$I_{MAX} = 2I_{C1} \left| \cos \frac{\theta_1 - \theta_2}{2} \right| \quad (18)$$

であることから、 I の最大電流は 2 つの接合の位相差の差に依存することが分かる。位相差の差を求めるために、超伝導体リングの中心での磁束の量子化について考える。また、リングの端から中心までの距離はロンドン侵入長よりも長いとする。超伝導リングの中心を積分経路として計算すると以下の式が成り立つ。

$$2\pi n\hbar = \frac{m}{n_S q} \oint \mathbf{J}_S d\mathbf{s} + 2q \oint \mathbf{A} d\mathbf{s} \quad (19)$$

ここで、 \mathbf{J}_S は超伝導電流密度、 m, q, n_S は超伝導電子対の質量と電荷、密度、 \mathbf{A} はベクトルポテンシャルである。超伝導リングの中心では完全反磁性により電流は流れないため、リングの中心では $\mathbf{J}_S = 0$ が成り立つ。その結果、2 項目の外部磁界による位相変化と磁界によるベクトルポテンシャルを通じた位相変化のみを考えることが出来る。

この場合、外部磁界による 2 つの接合の位相差成分は、

$$\hbar(\varphi_{1B} - \varphi_{1A}) - \hbar(\varphi_{2B} - \varphi_{2A}) \quad (20)$$

となる。式 19 において、Stokes の定理より、

$$\begin{aligned} 2q \oint \mathbf{A} d\mathbf{s} &= 2q \int_S (\nabla \times \mathbf{A}) d\mathbf{S} \\ &= 2q \int_S \mathbf{B} d\mathbf{S} \\ &= 2q\Phi \end{aligned} \quad (21)$$

となるので、

$$\hbar(\varphi_{1B} - \varphi_{1A}) - \hbar(\varphi_{2B} - \varphi_{2A}) - 2q\Phi = 2\pi n\hbar \quad (22)$$

という関係が得られる。 $\theta_1 = \varphi_{1B} - \varphi_{1A}, \theta_2 = \varphi_{2B} - \varphi_{2A}$ なので、式 22 は以下の式に変形できる。

$$\theta_1 - \theta_2 = \frac{2q}{\hbar}\Phi + 2\pi n \quad (23)$$

この式 23 より、式 18 で示した dc-SQUID の出力電流の最大値を以下のように記せる。

$$I_{max} = 2I_{C1} \left| \cos \left(\frac{\pi\Phi}{\Phi_0} \right) \right| \quad (24)$$

ここで、 $\Phi_0 = h/2q$ とおいた。

式 24 を図式すると、図 9 のようになる。

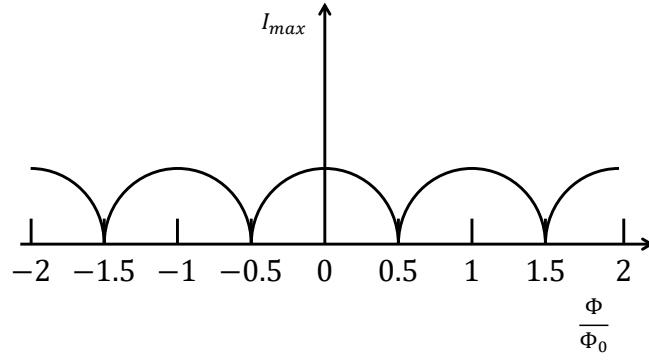


図9 dc-SQUID を流れる最大電流の磁束依存性

図からも分かるように、外部磁界の大きさが磁束量子 Φ_0 変化する毎に最大電流 I_{max} は周期的に変化している。本研究で設計する超伝導出力増幅回路は SQUID の臨界電流値を外部磁場によって変動させることで出力電圧を変化させている。

5 単一磁束量子回路

単一磁束量子 (SFQ: Single Flux Quantum) 回路は、図 (10) に示すようなジョセフソン接合を 2 つ含む超伝導ループで構成されている回路である。

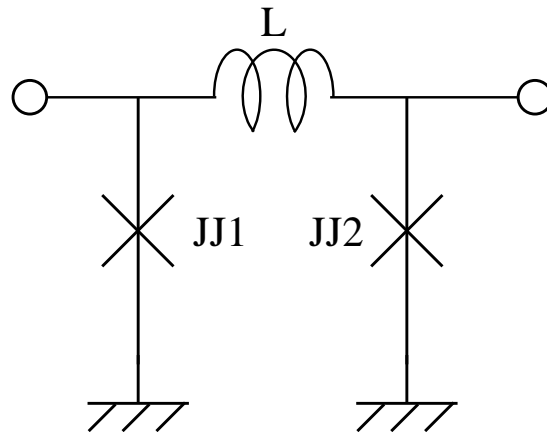


図 10 超伝導ループ

図中の × 印がジョセフソン接合である。構造としては上で述べた dc-SQUID と同様の構造と取っている。超伝導ループ内で磁束が量子化される現象を応用し、ループ内の磁束の有無を論理回路での"1"と"0"に対応させて演算を行うことで情報担体として扱う回路である。図 10 は最も基本的な SFQ 回路の構成であり、この構成以外にも 3 接合を有するループを用いる、インダクタ同士の磁気結合によってループ内の SFQ を制御する、インダクタの値と接合の臨界電流値との組合せなどの方法で SFQ の伝搬、増倍、消去等の様々な処理を実現する。この磁束量子の有無による演算方式は東北大学の中島康治氏が中心となって研究が進められた。[13] その後 Likharev らによって SFQ 論理ゲートとして体系化された。[4]

SFQ 回路は 100GHz を越える非常に高速な動作が可能かつ消費電力が非常に小さいという特徴も持っている。超伝導状態では電流による電圧降下が生じないため、電力を消費するのはジョセフソン接合がスイッチする時のみである。そのスイッチでの消費電力も論理ゲートあたりで 100nW 程度であるため、現在用いられている半導体と比較しても 3 桁程度小さくなる。[14]

以降は SFQ の伝搬方法や本研究で使用した論理ゲートに関する説明を行う。

6 ジョセフソン伝送線路

ジョセフソン伝送線路 (JTL: Josephson Transmission Line) は、以下の図 11 に示すようなジョセフソン接合を 2 つ含む超伝導リングを複数繋げた回路のことである。

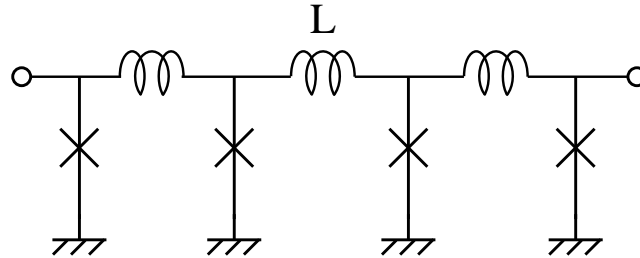


図 11 JTL の等価回路図

この回路は接合のスイッチで次のループへと SFQ を遷移する役割を持っており、SFQ 回路での基本的な回路構成である。

実際に次のループに遷移させるには SFQ を保持する電流に加えて接合に対してバイアス電流を加えている。このバイアス電流は SFQ を保持する電流との和が接合の臨界電流値を越えるように設定されており、単体ではスイッチしない。SFQ がループ内に入った際、SFQ の保持電流とバイアス電流との和が接合の臨界電流値 I_C を越える。結果、接合でスイッチが生じて SFQ が通りぬけられるようになり、SFQ の磁場の向きに応じたローレンツ力によって隣のループに遷移する。これを遷移先のリングでも同様に繰り返すことで SFQ を伝搬させていく。この流れを図にしたのが下の図 12 である。

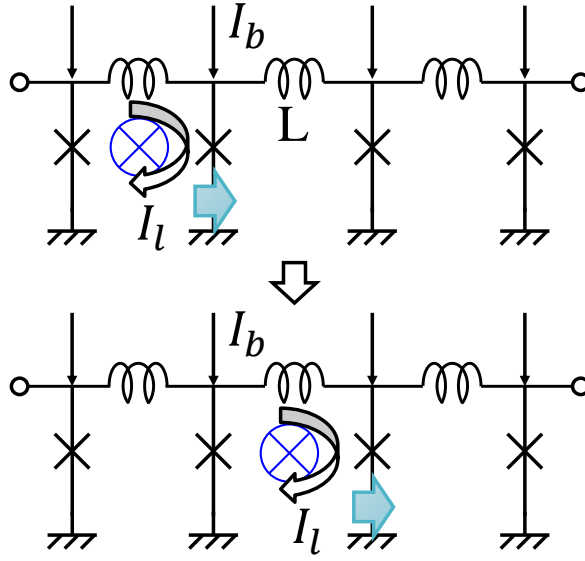


図 12 JTL による SFQ の伝搬

ジョセフソン接合を SFQ が通過する際、接合の両端の位相は 2π 変化する。同時に交流ジョセフソン効果の式で示される電圧も発生する。

$$V = \frac{\Phi_0}{2\pi} \frac{d\theta}{dt} \quad (25)$$

ここで、 θ はジョセフソン接合の位相差である。接合で発生する電圧はパルス状の波形となっているため、SFQ パルスという風に呼ばれる。そして SFQ 回路に周波数 f の SFQ パルス列を印加した時、ジョセフソン接合では以下の式 (26) に基づいて平均電圧が発生する。

$$\langle V \rangle = \Phi_0 f \quad (26)$$

ここで、 Φ_0 は定数であるため、発生する平均電圧は周波数にのみ依存する。現在ではセシウム原子を用いた周波数標準器を用いることで 15～16 桁という高い精度で周波数を発生させることが出来るため、ジョセフソン接合で発生する平均電圧も非常に高精度にすることが可能である [11, 15]。

また、ジョセフソン接合のスイッチング時間は 3 つの式があり、比較して大きい方をスイッチング時間としている。[16]

1 つはジョセフソン接合の抵抗 R とキャパシタンス C を用いて以下の式で示される。

$$\tau_C = RC \quad (27)$$

2 つ目はジョセフソン接合の抵抗 R とインダクタンス L を用いて以下の式で示される。

$$\tau_L = \frac{L}{R} = \frac{\Phi_0}{2\pi I_C R} \quad (28)$$

3 つ目は量子力学から考える。量子力学での不確定性原理では、運動量と位置の不確かさの関係が以下の式で示されている。

$$\Delta x \Delta p \geq \hbar \quad (29)$$

ここで、 $\Delta x, \Delta p$ はそれぞれ位置と運動量の不確かさである。この関係は時間とエネルギーの関係でも同じことが言え、それぞれ $\Delta t, \Delta E$ と置き換えることが出来る。

$$\Delta E \Delta t \geq \hbar \quad (30)$$

また、ジョセフソン接合が電圧状態の時の電圧と、超伝導電流のエネルギーは以下の式 (31)、(32) で示される。

$$V = I_C R \quad (31)$$

$$\Delta E = 2qV \quad (32)$$

ここで I_C は接合の臨界電流である。この 2 式を式 (30) に代入することでジョセフソン接合のスイッチング時間を求めることが出来る。

$$\tau = \frac{\Phi_0}{2\pi I_C R} \quad (33)$$

なお、実際の回路においては、式 (27) または (28) が支配的となるため、追加で以下の制約を満たす必要がある。

$$\beta_C = \frac{\tau_C}{\tau} = \frac{2\pi I_C R^2 C}{\Phi_0} \quad (34)$$

この式はマッカンバーパラメータと呼ばれており、このパラメータによって接合の I-V 特性は大きく変化する。このパラメータを用いることによって R と L を用いたスイッチング時間には以下の関係が示せる。

$$\tau_C = \beta_C \tau_L \quad (35)$$

SFQ 回路での高速化ではスイッチング時間を短くしつつ、式 (34) を満たす必要がある。しかし、ジョセフソン接合のみでは融通が効かないため、接合と並列にシャント抵抗と呼ばれる並列抵抗を付けることで抵抗値を調整している。この場合でもスイッチング時間にかかる R はシャント抵抗を含めた値として扱うことが出来る。現在では $\beta_c = 1$ の時に R, L のスイッチング時間が等価になるため、SFQ 回路で最も最適な値であると考えられている。

7 SFQ 論理セル

上で述べた RSFQ 回路の応用として RSFQ 論理回路が挙げられる。上でも簡単に述べたが、SFQ 論理回路は SFQ 回路上で論理回路を表現する体系であり、1991 年に Likharev らによって体系化された [4]。SFQ 論理回路での動作は磁束量子を単位として動作するが、どのタイミングで入力されたかを明確にするためにクロック信号による同期が取られている [17]。本研究では別途クロック信号を扱うような回路を構成していないため、クロックを考慮せずに説明を行う。

7.1 Splitter

Splitter 回路は入力された SFQ を 2 方向に分配する出力論理回路である。図 13 に等価回路図を、図 14 に実際に動作させた時の入出力結果を示す。

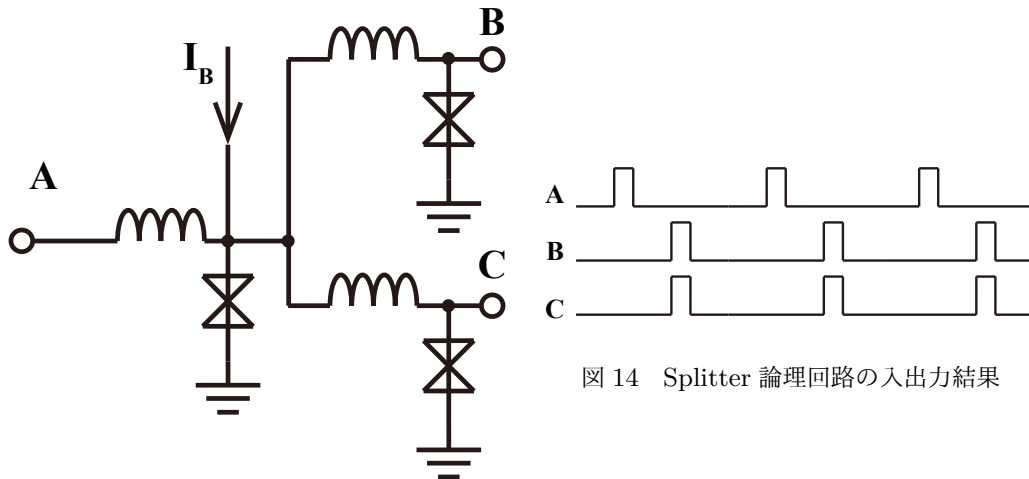


図 13 Splitter 論理回路の等価回路図

一般的な回路とは異なり、ただ 2 つに配線を分岐しただけでは動作しない。これは、回路の担体が SFQ であるため、左右に SFQ を分離させるための接合を用意しておく必要があるためである。

SFQ はどちらか片方のみ出力されることは無く、両方から同時に出力される。これは JTL の先を 2 つの JTL に分配しているため、それぞれの JTL に SFQ がスイッチして伝搬するためである。JTL は図 13 でいう入力 A から出力 B と C の間にある 2 接合ループを指している。

7.2 Confluence Buffer

Confluence Buffer 回路は 2 方向から入力された SFQ を 1 つの方向に結合する出力論理回路である。図 15 に等価回路図を、図 16 に実際に動作させた時の入出力結果を示す。

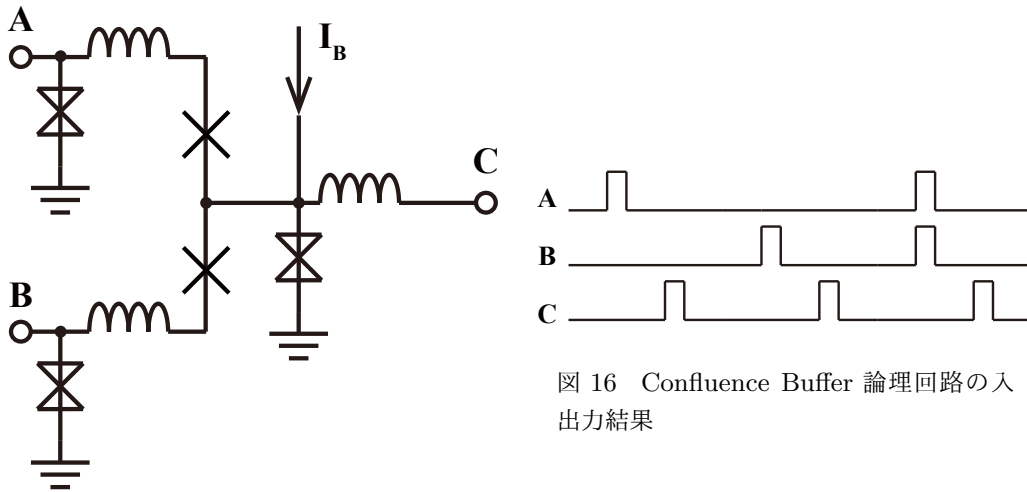


図 15 Confluence Buffer 論理回路の等価回路図

図 16 Confluence Buffer 論理回路の入出力結果

2 方向からの SFQ 入力はある一定以上の時間差を空けて入力する必要がある。同時に近いタイミングで入力されると、2 方向からの電流が干渉することで 1 つだけ SFQ が出力されるなどの誤動作を引き起こす。また、この回路には入力側の 2 つのループ上に接合が存在する。これは入力側からの SFQ が別の入力側に出力されないようにするための接合であり、エスケープ接合と呼ばれている。このエスケープ接合でスイッチすると SFQ はループの外に放出されるため、ループ側から見ると次のループにスイッチせずにそこで SFQ が消滅するように見える。もしこのエスケープ接合が無い場合、回路の動作としては上述した Splitter と全く同じ動作になる。エスケープ接合は入力側の接合の臨界電流値よりも低く設定されている。この臨界電流値の差により、入力から来た SFQ はエスケープ接合でスイッチせずに通り抜け、もう片方の入力からきた SFQ はより臨界電流値の少ないエスケープ接合でスイッチして回路の外に放出される。

7.3 TFF

TFF(T-flip flop) 回路は SFQ の入力毎に前の出力状態と反対の出力をする論理回路である。相補出力をするため、両方の出力からは交互に逆向きの SFQ が出力される。図 17 に回路の等価回路図、図 18 に

回路の動作例を示す。

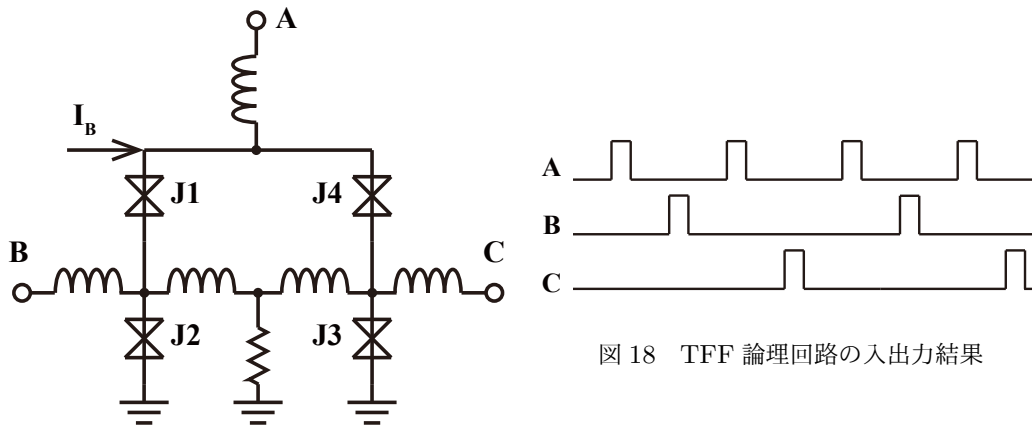


図 17 TFF 論理回路の等価回路

図 18 TFF 論理回路の入出力結果

図 17 を用いて動作の説明を行う。最初の SFQ が A から入力されると、J1~4 のループに環状電流が流れる。この電流とバイアス電流により、J2 と J4 がスイッチする。J2 と J4 がスイッチすることにより、各接合の両端に SFQ が生成され、J2 の左側に生成された SFQ は出力 B から伝搬していく。J4 で生成された SFQ は片方の SFQ が J1 と J4 のループに、J2 で生成された右側の SFQ は J2 と J3 のループに蓄えられる。次の SFQ が入力されると、蓄えられている SFQ による環状電流によって J1 と J3 がスイッチする。これにより接合の両端には SFQ が生成され、その SFQ は蓄えられている SFQ とは逆向きの磁束を持つため打ち消しあう。また J3 で生成された出力 A 側の SFQ は出力 A へ向けて伝搬する。よって最初の SFQ 入力の前の状態に戻り、以後はこの 2 パターンをループし続ける。

どちらの出力から先に出るかは決まっており、図 17 の場合で出力 B から先に出力される。また、リセット機構を備えている TFF 回路もあり、この場合はリセットに SFQ が入力されると状態に関係なく次の出力が出力 B に初期化される。本研究では SET と RESET の 2 方向から SFQ を交互に入力するために TFF 回路を使用している。

7.4 NDRO

NDRO 回路は Non-Destructive-ReadOut 回路の略であり、信号の出力時にその状態を保持することが可能なレジスタ回路である。図 19 に回路の等価回路図、図 20 に回路の動作例を示す。

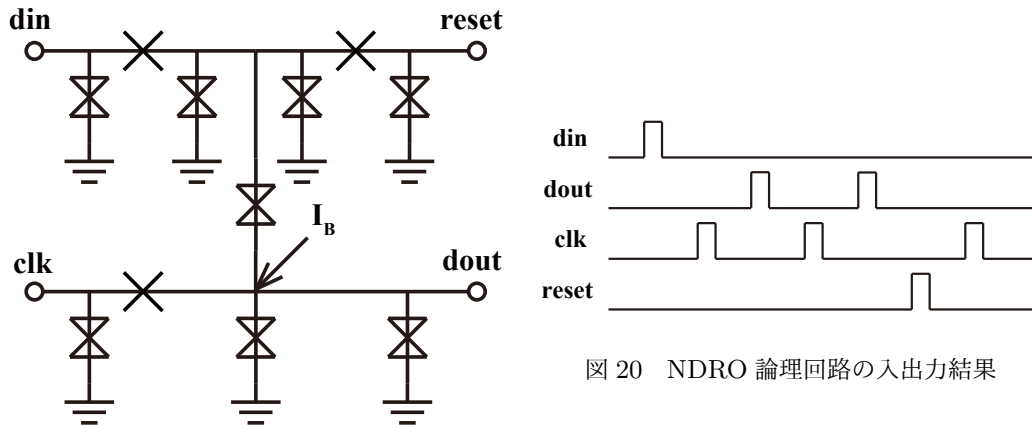


図 20 NDRO 論理回路の入出力結果

図 19 NDRO 論理回路の等価回路

din から入力された SFQ は din と reset の間の配線の中点が下に繋がっているループに蓄えられる。蓄えられたループは下の接合にも電流が流れるため、バイアス電流と合わさって臨界電流値に近い電流が流れている状態となる。この状態で clk から SFQ を入力すると、そのループ電流の分も加わるので接合がスイッチし、dout から SFQ が出力される。SFQ が蓄えられている状態で reset から SFQ が入力されると、先程とは逆向きのループ電流が流れるためループの SFQ は消滅する。これにより clk が入力されても接合の臨界電流値を超えないため、dout には何も出力されない。din、reset、clk の各入力のループ上にある抵抗は SFQ の逆流を防ぐためのエスケープ接合である。一度 din に SFQ を入力すれば reset するまで clk に SFQ を入れることで dout から SFQ を出力させ続けることが可能なため、本研究では dout と clk の間を遅延時間を調整したループを設置し、SFQ 発振回路として使用している。

7.5 dc/SFQ

dc/SFQ 回路は入力された電流がある閾値を越えている場合に SFQ を生成する論理回路である。図 21 に回路の等価回路図、図 22 に回路の動作例を示す。

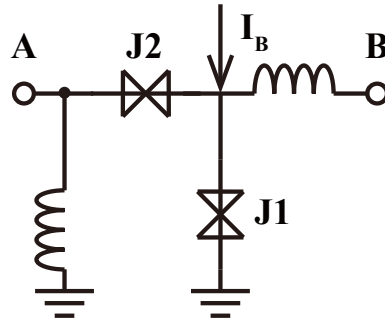


図 21 dc/SFQ 論理回路の等価回路

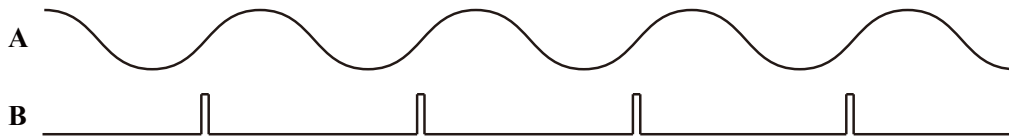


図 22 dc/SFQ 論理回路の入出力結果

等価回路図ではクロック入力が無い場合の回路を載せているが、クロックとの同期を取るためにクロック入力があった場合に初めて SFQ が出力されるように調整された dc/SFQ も存在する。本研究で設計した回路は外部クロックによる同期は使用していないので、図 21 に示すような構成の dc/SFQ を使用している。dc/SFQ にある閾値以上、つまり接合の臨界電流値以上の電流を流すと図 21 の接合 J1 がスイッチする。SFQ が何も無い状態で接合がスイッチすると接合を挟んで左右にそれぞれ磁束の向きが逆向きの SFQ が生じる。これは回路全体で磁束の状態を一定に保つために生じる。接合 J1 の右側に生じる SFQ はそのまま回路中を伝搬していくが、左側に出来た逆向きの SFQ はそのままだと入力を逆流してしまうので、接合 J2 をスイッチして回路の外に放出されるようになっている。接合 J2 の前に設置している GND に繋がるインダクタンスは、接合 J1、J2 を含むループを作ることによってそのループ内に SFQ が出来るようにするために設置している。このインダクタンスが無いと接合 J1 と J2 を含むループは入力 A の前の回路を含むループとなり、そこに SFQ が生じると前の回路に悪影響を及ぼす可能性があるためである。

7.6 SFQ/dc

SFQ/dc 回路は SFQ パルスを電圧に変換する回路である。図 23 に回路の等価回路図、図 24 に回路の動作例を示す。

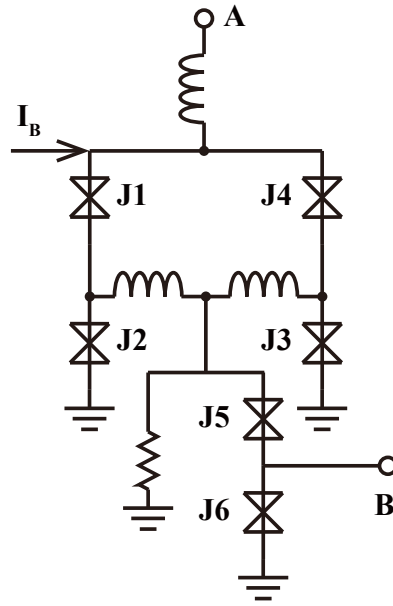


図 23 SFQ/dc 論理回路の等価回路

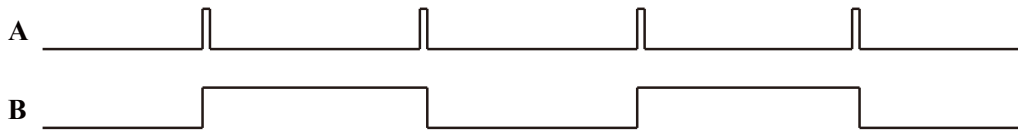


図 24 SFQ/dc 論理回路の入出力結果

等価回路図は TFF をベースにした回路であるが、RSFF をベースにした回路も存在する。本研究では TFF ベースの SFQ/dc 回路を使用しているため、以降では TFF ベースで話を進める。

SFQ 入力時の動作は TFF 論理回路と同様であるが、出力が電圧振幅となるように設計されている点が異なっている。SFQ の入力毎に電圧振幅が切り替わり、出力周期は SFQ/dc への入力周期の半分になる。

接合 J5 と J6 は NDRO 論理回路の等価回路図 (図 19) と似た役割を持つ。J2 と J3 のループに SFQ が蓄えられると、J5 と J6 にも電流が流れるようになる。ここに J6 にだけ流れるようにバイアス電流を加えておくと、J6 がスイッチするが、J5 がスイッチしないため、J6 は次の SFQ パルスの到達まで電圧状態を維持する。この間は J6 を挟んだ出力端子と GND の間で電位差が生じるため、出力には電圧が生じ続ける。

次の SFQ が到達すると、J6 は既に電圧状態であるため、J5 に電流が流れて電圧状態になる。すると

J5 を通った電流は並列に設置した抵抗を通るため、J6 はゼロ電圧状態に戻る。よって出力端子と GND との電位差が消え、出力にも電圧は生じなくなる。回路の出力は NRZ 信号として振舞うため、出力信号の周期は入力周期の半分となる。

8 超伝導出力増幅回路

SFQ 回路は情報担体として SFQ を扱うため、回路の出力も SFQ となる。この SFQ パルス 1 つあたりのエネルギーは $J = I_C \Phi_0$ で求められる。ここで、 I_C が接合の臨界電流値、 $\Phi_0 = h/2e = 2.05 \times 10^{-15}$ が磁束量子の値であり、定数値である。一般的な接合の臨界電流値は数百 μA オーダーとなるので、SFQ パルスのエネルギーは 10^{-19}J オーダーとなる。例えば、 $I_C = 100\mu\text{A}$ の場合、 $J = I_C \Phi_0 = 2.05 \times 10^{-19}\text{J}$ オーダーとなる。極低温でのデータ転送であればあまり問題にはならないが室温回路への転送を考えた場合、熱雑音などの影響を大きく受けてしまい、正常な転送が出来ない。そのため、SFQ パルスのある程度の大きさまで電圧パルスとして増幅する増幅回路が必要となる。

RSFQ 出力増幅回路には様々な種類の回路が考案されている。以下に今までに考案された回路のバイアス方式、最大出力振幅、最大動作周波数を表 1 に示す。

表 1 Comparison of RSFQ output amplifiers.

	Suzuki stack [18]	4JL gate [19]	HUFFLE [20]	SQUID array [21]	Dist. Amp. [22]
Bias type	AC	AC	DC	DC	DC
V_{max} [mV]	60	2.3	1.5	3	1.75
f_{max} [GHz]	1	2	4	120	10
Nb tech. [kA/cm ²]	4.5	2.5	1.0	1.5	4.5
reference	[23]	[24]	[25]	[26]	[22]

ここでは本研究で使用した RSFQ 分布型アンプ (Distributed Amplifier) について説明する。

8.1 RSFQ 分布型アンプ (RSFQ Distributed Amplifier)

RSFQ 分布型アンプは高周波アンプとして使用されている分布型アンプを超伝導回路に応用したものである。この回路は元々真空管時代に考案された回路であり [7]、現在では伝送線路とトランジスタを組み合わせることで寄生容量を削減、帯域幅と利得の両方の向上を可能にする回路として用いられている [8]。2010 年に Herr によって RSFQ 分布型アンプが考案され [22]、12 段分布型アンプが作製された。

Herr らの結果では、SFQ パルスの入力周期が 10Gbps で最大出力振幅が 1.75mV という結果が得られた。また、最大出力振幅が得られた領域は、動作領域の中心から $\pm 12\%$ であった。図 25 に先行研究でのバイアス電流に対する出力電圧振幅特性、図 26 に 10Gbps 入力時のアイパターンを示す。

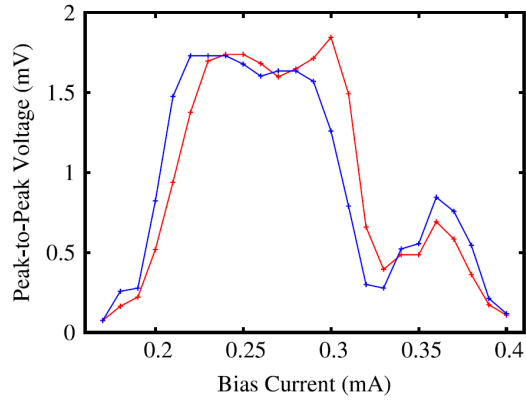


図 25 バイアス電流に対する出力電圧振幅特性。横軸がバイアス電流で単位は mA、縦軸が出力電圧振幅で単位は mV である。

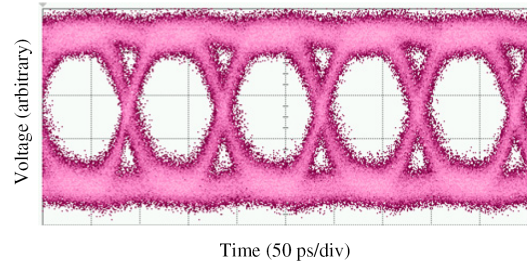


図 26 10Gbps 入力時のアイパターングラフ。横軸は時間軸であり 1 メモリで 50ps である。

図 27 に回路の等価回路図、図 28 に回路の動作例を示す。

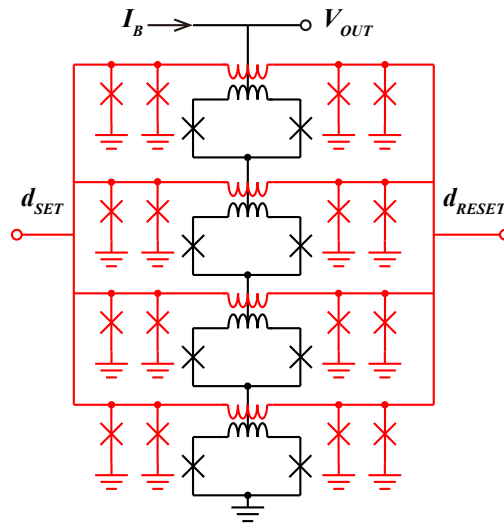


図 27 RSFQ 分布型アンプの等価回路図

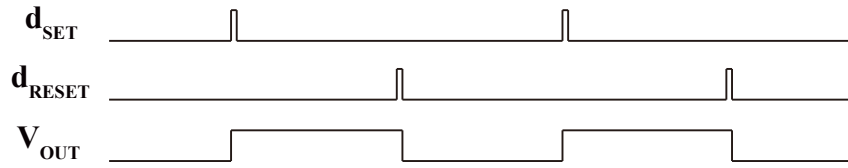


図 28 RSFQ 分布型アンプの動作例

分布型アンプは超伝導回路では SQUID とそれに結合するストレージループから 1 段が構成されており、ストレージループに蓄えられる SFQ の有無によって SQUID を制御し、出力電圧が変化させる。ストレージループには SET 方向と RESET 方向の 2 方向から SFQ が入力され、今回の設計では TFF を通して SFQ パルス毎に 2 方向を切り替えている。

RSFQ 分布型アンプは複数段を直列に接続することで出力電圧を増幅しているが、各段に入力する SFQ は JTL と Splitter によって分配されるため、次の段への SFQ の伝搬に遅延時間が生じる。また、出力回路をそのまま室温機器で観測する場合はインピーダンス整合を取る必要がある。超伝導体のインピーダンスは数 Ω と非常に低く、このままでは室温機器と接続した場合の反射波の影響が大きく表われてしまうためである。本研究では高速測定用の回路において、現在の同軸ケーブルや測定機器で使用されているインピーダンス 50Ω に回路上で整合を取るよう設計を行なった。これらの整合は回路だけで取るのは困難なため、各段に LC 共振回路を挿入することで整合を取る方法を採用している。詳細な設計値などについては後述する。

8.2 RSFQ 分布型アンプの動作原理

動作原理の説明に際し、以下の 3 つの仮定をする。

- 回路内の SQUID には外部からはなにも磁場が印加されておらず、電圧状態に遷移もしていない状態とする。
- 回路の出力から印加するバイアス電流値は回路動作に適切な値であるとする。
- 入力 SFQ は TFF を通して 2 方向に分岐するという方法で入力するものとする。

8.2.1 SET 方向から SFQ が入力された場合

最初に入力された SFQ は、TFF を通して図 29 に示すように SET 方向からストレージループに到達する。図 29 は SET 方向から入力された場合の動作を示している。図 29 では 4 段 RSFQ 分布型アンプの等価回路図を載せているが、1 つの SFQ 入力で JTL と Splitter を通して全ての段のストレージループに SFQ は入力される。

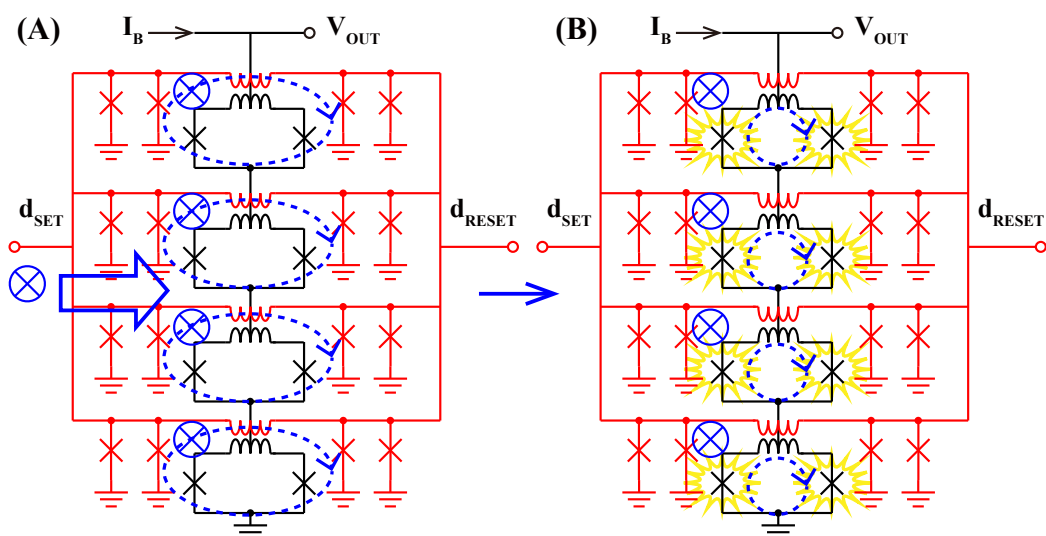


図 29 SET 方向からストレージループに SFQ が入力される場合の動作

ストレージループに到達した SFQ はそのループ内に蓄えられる。これは、他の JTL と違いストレージループの接合間インダクタンスを大きく設定しているからである。ストレージループに SFQ が蓄えられ、そのループには環状電流が流れる。この環状電流の流れる向きは SFQ の向きと対応しており、図 29 の場合だと手前から奥に磁束が流れることからアンペールの法則により時計回りに電流が流れる。ここで、下の SQUID とは磁氣的に結合しているため、下の SQUID に対しても誘導電流が生じる。SQUID に生じる誘導電流はストレージループと同様に循環電流となるが、ここで SQUID の出力からバイアス電流を流すと循環電流とバイアス電流の和によって SQUID 全体の臨界電流値が減少する。これにより、通常より少ないバイアス電流で SQUID が電圧状態に遷移する。

8.2.2 RESET 方向から SFQ が入力された場合

次に入力される SFQ は TFF を通して RESET 方向からストレージループに到達する。図 30 に RESET 方向から入力された場合の動作を示す。

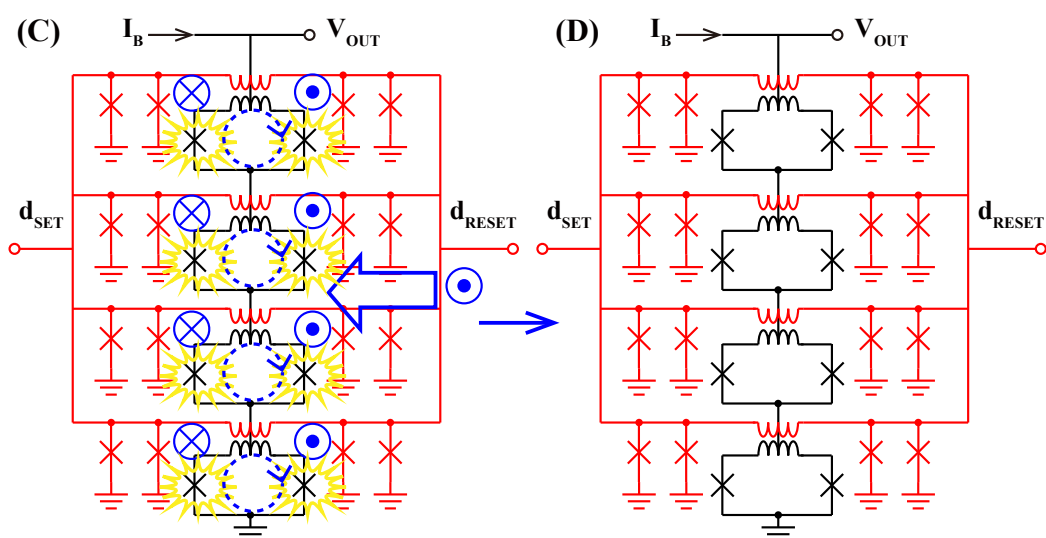


図 30 RESET 方向からストレージループに SFQ が入力される場合の動作

この時、SET とは逆方向から SFQ が入力されるため、ストレージループに蓄えられている SFQ とは磁束の向きが反対になる。この状態でストレージループに到達すると逆向きの SFQ 同士によってストレージループ内の SFQ は対消滅し、ストレージループには何も無い状態になる。循環電流も SFQ の消滅により消えるので、SQUID を流れる誘導電流も消える。よって、SQUID の臨界電流値は元の電流値に戻り、SQUID はゼロ電圧状態に戻る。

回路の出力電圧は入力信号毎に変化しているため、Non-return-to-zero(NRZ) 方式で出力が現れる。また 2 個の入力 SFQ 毎に出力信号が切り替わるため、回路の出力周期は SFQ の入力周波数の半分で出力される。

第 III 章

回路設計及び試作方法

9 Nb 集積回路作製プロセスの設計仕様

本研究での回路の作製は (国) 産業総合技術研究所 (AIST) の超伝導クリーンルーム (CRAVITY) において、Nb 標準プロセス (AIST-STP2) を用いて作製された [27, 28]。以下にデバイスの断面構造を示す。

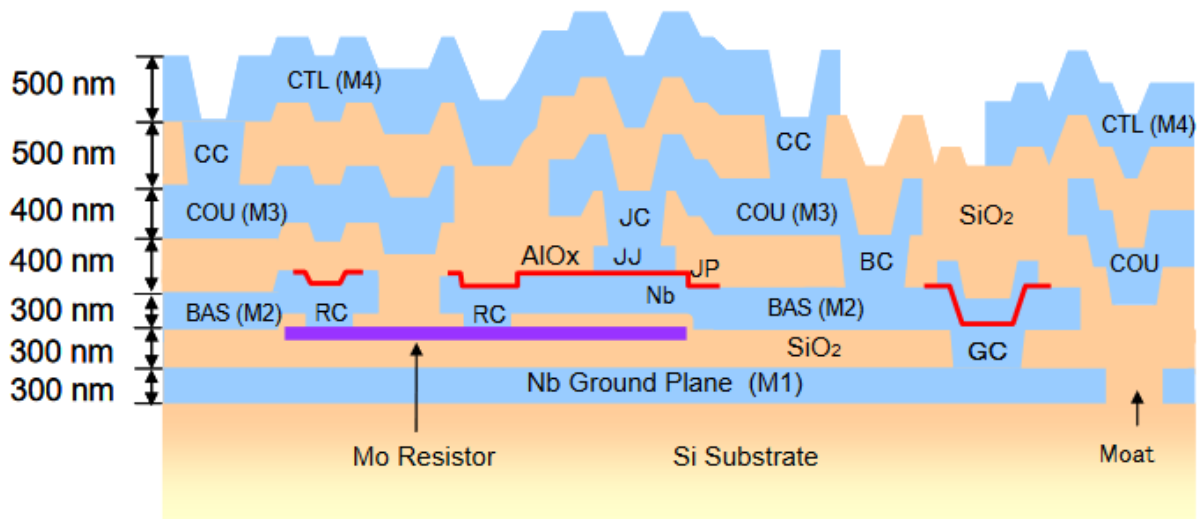


図 31 Nb 標準プロセスのデバイス構造 [27]

次に以下の表 2 に Nb 標準プロセスの各レイヤー構成の詳細を示す。

表 2 Nb 標準プロセスのレイヤー構成

レイヤー名	機能	材料	膜厚 [nm]
GP	グランド面	Nb	300
RES	抵抗 (Mo)	Mo	80
RC	RES/BAS 間コンタクト		
GC	GP/BAS 間コンタクト		
BAS	接合の下部電極及び下部配線	Nb	300
JP	接合のプロテクション	Al, AlO _x	
JJ	接合	Nb	150
BC	BAS/COU 間コンタクト		
JC	JJ/COU 間コンタクト		
COU	接合の上部電極及び上部配線	Nb	400
CC	COU/CTL 間コンタクト		
CTL	最上部の配線又はシールド層	Nb	500

また, Nb 標準プロセスの設計上のルールを以下の表 3 に示す。

表 3 Nb 標準プロセスの設計ルール

最小線幅	1.5 μm
最小スペース	1.0 μm
最小接合サイズ	2.2 μm 角 (実サイズ 2.0 μm 角)
接合の縮小値	0.2 μm
接合の臨界電流密度	2.5kA/cm ²
シート抵抗	1.2 Ω
最小コンタクトサイズ	1.0 μm 角
アライメントマージン	0.5 μm
BAS の縮小値	0.2 μm
COU, CTL の縮小値	0
RES の縮小値	0

AIST の CRAVITY の中ではこれ以外にも臨界電流密度が 4 倍の 10kA/cm² の HSTP[29] や、同じく臨界電流密度が 4 倍かつ CMP を用いて 9 層に層が増加した ADP2[30] などが存在する。超伝導分布型アンプの先行研究 [22] で使用されているプロセスはアメリカ Hypres 社のプロセスであり、臨界電流

密度は $4.5\text{kA}/\text{cm}^2$ である。臨界電流密度が変化に対してスイッチング速度はその平方根に反比例することが知られている [31]。そのため例えば我々の使用する STP2 と Hypres Nb $4.5\text{kA}/\text{cm}^2$ プロセスを比較した場合、スイッチング時間は Hypres 側よりも $\sqrt{2.5/4.5} \simeq 0.75$ 倍だけ遅くなる。

10 SFQ 論理セルライブラリ (CONNECT Cell Library)

我々が研究している RSFQ 回路は 7 節で述べたように、JTL が基本単位となっている。この JTL や TFF などといった論理セルも、本来であれば自ら設計をする必要があるが、全て設計していると非常に手間がかかる、回路が動作しない場合のアンプ部分とその他の部分との切り分けが困難という問題点がある。そこで本研究では分布型アンプ以外の入力部分や SFQ パルスを生成する部分などは CONNECT Cell Library と呼ばれるセルライブラリを利用した [32]。

このライブラリは NEC や名古屋大学などが共同で作製したライブラリで、我々が使用している AIST-STP2 プロセス用に作製されている。作製されたセルは JTL、Splitter などの SFQ 伝搬に関するセルや、AND、OR、NOT、TFF などの論理セルなどの複数種類のセルが設計されている。これらのセルをベースに組合せていくことで、用意に RSFQ 論理回路を設計することが可能になる。また、いずれのセルもバイアス電流に対する動作領域が設計値の $\pm 30\%$ までの範囲で動作することを保証している。セルのサイズは $40 \times 40 \mu\text{m}$ のサイズが基本形であり、それ以外のサイズのセルも整数倍のサイズになるように設計されている。本研究ではこのセルライブラリのサイズに合わせて分布型アンプの各段を設計した。

11 超伝導集積回路のシミュレーション

回路の設計において JSIM [33] による超伝導回路のシミュレーションを行い、SCOPE2 [34] を用いて回路の動作領域の計算を行った。JSIM はカリフォルニア大学バークレー校で作成されたジョセフソン接合を用いた超伝導回路のシミュレータである。JSIM は汎用回路シミュレーションソフトである SPICE[35] と同じ使い勝手で使えるように作製されており、ジョセフソン接合のシミュレーションは RCSJ モデルを使用している。バイアス電流を一定値ずつ変化させながら計算するなどの機能を JSIM は備えておらず、また計算結果も位相の変化から電圧値などを求めるため、自動的にパラメータを変更する処理、計算結果を用いたデータ処理などは自作のスクリプトで処理を行った。言語は Python3 を用いた。具体的なコードなどは付録に記載する。SCOPE2 はその JSIM を用いてパラメータ変化させていった場合の動作領域を求めるソフトである。あらかじめ接合の最大スイッチ間隔、つまり次の接合に到達するまでに最大でどれぐらいの遅延時間を許容できるかを SFQ の流れに沿って記した判定ファイルを用意し、それを SCOPE2 がパラメータを変更した際にどこまで満たすかを回路の動作領域として出力する。

本研究では二重 SQUID 構造分布型アンプの設計、動作速度、単一 SQUID 構造分布型アンプとの比較に使用している。

12 作製回路のレイアウト設計

12.1 低速測定

本研究で作製した分布型アンプは 10 節でも述べたようにセルライブラリベースで作製しており、同時にステージ間の整合回路もセルで作製している。設計回路によってセルのサイズは異なるが、低速測定では SQUID とストレージループからなる各段と間に挟む整合回路は共に $80 \times 80 \mu\text{m}$ 、高速測定では各段と整合回路を両方合わせて $80 \times 80 \mu\text{m}$ として設計した。図 33、34、35 に作製した 4、12、24 段二重 SQUID 構造分布型アンプ全体の顕微鏡写真を、図 32 にアンプ 1 段を拡大した顕微鏡写真を示す。各図には縮尺が示しており、図 32 には各端子が回路のどこに対応するかを示している。

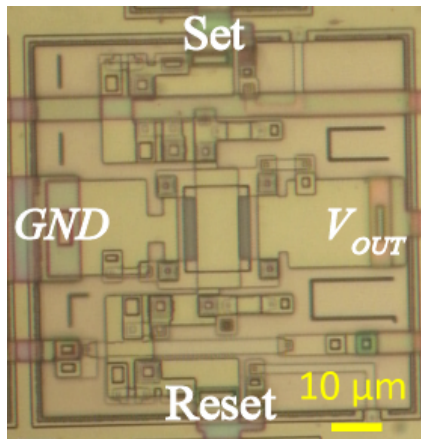


図 32 二重 SQUID 構造分布型アンプ 1 段の顕微鏡写真

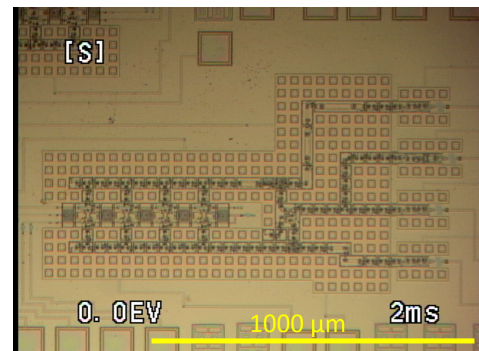


図 33 4 段二重 SQUID 構造分布型アンプ全体の顕微鏡写真

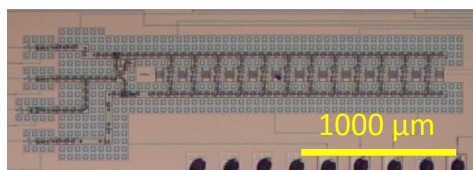


図 34 12 段二重 SQUID 構造分布型アンプ全体の顕微鏡写真

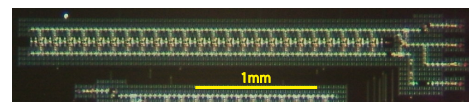


図 35 24 段二重 SQUID 構造分布型アンプ全体の顕微鏡写真

回路は入力回路と増幅回路部分に分かれており、入力回路は FG からの正弦波入力から dc/SFQ セルを通じて SFQ パルスを作成する。SFQ パルスは TFF セルによって 2 方向からストレージループに入力

され、結合する SQUID の臨界電流値を変動させる。一方増幅回路部分はストレージループと結合する SQUID と入力回路の SFQ 伝搬遅延と室温回路とのインピーダンス整合を取るための整合回路の 2 つからなる。整合回路は LC 回路で構成されており、L と C の値を調整することで次の段への SFQ パルスの伝搬遅延と室温回路のインピーダンス整合を兼ねている。

また改良した分布型アンプは SQUID の数が 2 個になっている [36]。2 個の SQUID はストレージループと結合するインダクタが共有されており、SQUID のパラメータを調整することでストレージループに蓄えられた SFQ に対して同時に SQUID の臨界電流値が変動するように調整している。

個々の SQUID には 2 個のジョセフソン接合が存在し、片方の接合にのみ並列にシャント抵抗が備わっている。これは両方シャント抵抗にするよりも接合のスイッチング速度を早め、より高い出力電圧を得ることができるためである。

また、従来回路からの変更点として、ストレージループの手前の JTL にエスケープ接合を設置した。これは複数回 SFQ が入力された時にストレージループ内に 2 つ以上の SFQ が入らないようにするために設置している。

単一 SQUID 構造分布型アンプ 1 段あたりの出力は今回の設計では 0.3mV と仮定した。SQUID が 2 個ある二重 SQUID 構造分布型アンプでは 1 段あたり 0.6mV となる。測定結果で計算するバイアス電流に対する動作領域はこの想定電圧を基準として動作領域を計算している。

今回設計した単一 SQUID 構造分布型アンプと二重 SQUID 構造分布型アンプの各段の等価回路図とパラメータを図 36 と図 37 に示す。

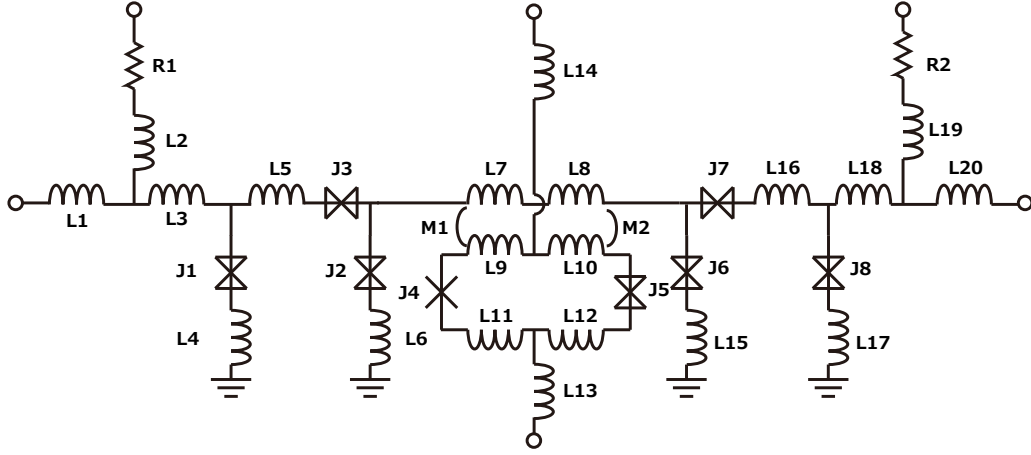


図 36 単一 SQUID 構造分布型アンプの 1 段分の等価回路図。パラメータは後述する JSIM と SCOPE2 を用いて求めた; $L1 = L4 = L15 = L17 = L20 = 0.50$ pH, $L2 = L19 = 3.50$ pH, $L3 = 2.00$ pH, $L5 = 2.58$ pH, $L6 = 0.46$ pH, $L7 = L8 = 5.70$ pH, $L9 = L10 = 4.70$ pH, $L11 = L12 = 0.32$ pH, $L13 = 2.40$ pH, $L14 = 1.60$ pH, $L16 = 4.00$ pH, $L18 = 3.00$ pH, $J1 = J8 = 0.22$ mA (1.71Ω), $J2 = J6 = 0.13$ mA (2.83Ω), $J3 = J7 = 0.20$ mA (1.84Ω), $J4 = 0.14$ mA, $J5 = 0.14$ mA (2.66Ω), $R1 = 12.92 \Omega$, $R2 = 15.66 \Omega$, $M1 = M2 = 3.62$ pH ($k = 0.70$).

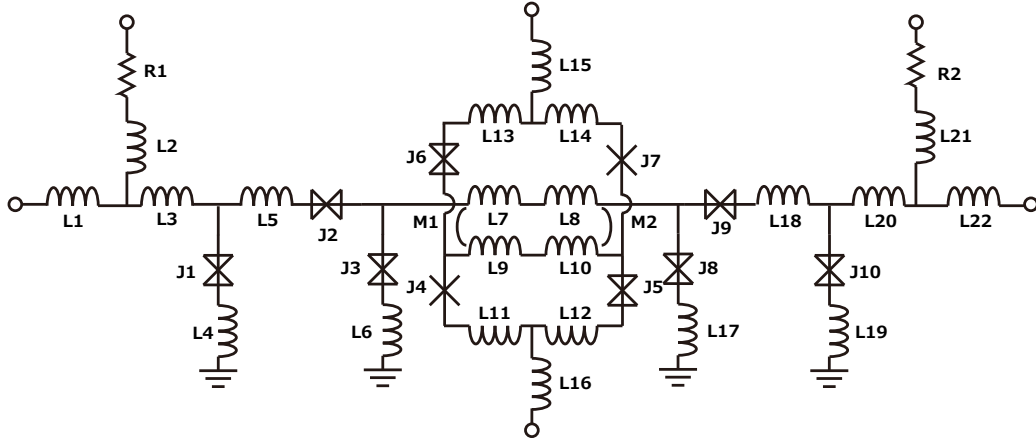


図 37 二重 SQUID 構造分布型アンプの 1 段分の等価回路図。パラメータの求めかたは単一 SQUID 構造分布型アンプと同様である; $L1 = L4 = L17 = L19 = L22 = 0.50$ pH, $L2 = L21 = 3.50$ pH, $L3 = L18 = L20 = 2.00$ pH, $L5 = 2.38$ pH, $L6 = 0.46$ pH, $L7 = L8 = 4.09$ pH, $L9 = L10 = 4.12$ pH, $L11 = L12 = 1.80$ pH, $L13 = L14 = 1.20$ pH, $L15 = 1.60$ pH, $L16 = 2.40$ pH, $J1 = J10 = 0.22$ mA (1.71Ω), $J2 = J9 = 0.20$ mA (1.84Ω), $J3 = J8 = 0.13$ mA (2.83Ω), $J4 = J7 = 0.10$ mA, $J5 = J6 = 0.10$ mA (3.73Ω), $R1 = 17.27 \Omega$, $R2 = 20.37 \Omega$, $M1 = M2 = 3.20$ pH ($k = 0.73$).

図 37、36 において、蝶ネクタイ型の記号がシャント抵抗が並列に接続されたオーバードンプ接合、×

印の記号がジョセフソン接合のみのアンダーダンプ接合を表している。接合パラメータの後ろにある抵抗は接合と並列に接続するシャント抵抗である。また図では端子は6つ存在するが、実際にセルに入出力する端子は4つであり、R1とR2に接続される端子は接合に印加するバイアス電流用の端子である。レイアウト作製時にはInductEX[37]を用いた数値計算を利用して作製を行った。

JSIMとSCOPE2を用いたパラメータの最適化では、SQUIDの LI_C 積を最小化するように最適化を行った。 LI_C 積を最小化することでパラメータの変化に対する動作マージンが増加し、より広い動作マージンを確保することが出来るためである。 LI_C 積ではSQUIDのインダクタンス L (図37の $L9, L10$)と接合の臨界電流値 I_C (図37の $J4 \sim J7$)の両方が関係してくるが、今回は I_C の値をプロセスで保証されている最低値である0.10mAに固定し、インダクタンスを変化させることによって最適化を行った。同様に、SQUIDとストレージループ間の結合係数も動作マージンに影響するため、最大化するように最適化を行った。実際の設計では、回路の下層に存在するGP層を取り除くことでより高い結合係数を実現している。

12.2 高速測定

高速測定で使用したパラメータは概ね低速測定と一致している。変更したパラメータを表4に示す。パラメータは図37に対応している。

表4 高速測定でのパラメータの変更点

パラメータ	$L5$	$R1$	$R2$
低速測定 (変更前)	2.38 pH	17.27 Ω	20.37 Ω
高速測定 (変更後)	2.28 pH	14.84 Ω	22.82 Ω

パラメータ以外では、各段の間に設置しているLC共振回路を用いて室温回路とのインピーダンス整合とSFQの伝搬遅延時間を考慮している。それに合わせて入力側のSFQの伝搬時間も共振回路の遅延時間と一致するように変更した。変更前と変更後の入力側JTLのレイアウト図を図38と39に示す。

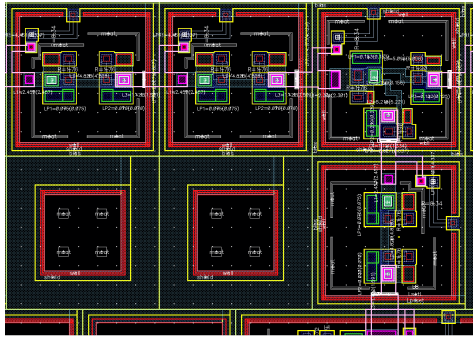


図 38 低速測定用回路での入力側 JTL レイアウト図

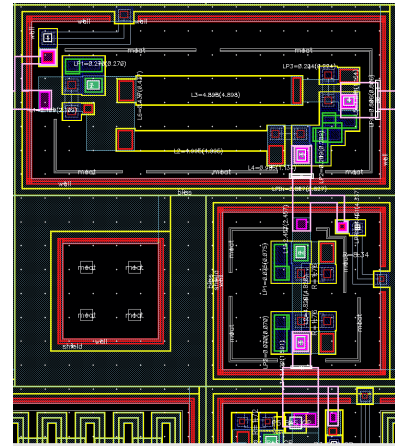


図 39 高速測定用回路での入力側 JTL レイアウト図

高速測定では低速測定の JTL2 つ、Splitter(SPL)1 つの構成を JTL と SPL を組み合わせたセル 1 つの構成に変更した。変更前の次段への遅延時間は 17ps であるが、変更後は 10ps まで減少した。遅延時間を減少する方向に変更したのは、後述する LC 共振回路の設計において遅延時間が大きいと専有面積が大きくなり、設計が難しくなるためである。また、遅延時間を減少することで段数を増やしても広い帯域幅を維持することが出来るという利点がある。

整合後の LC 共振回路のパラメータを図 40 に、その顕微鏡写真を図 41 に示す。

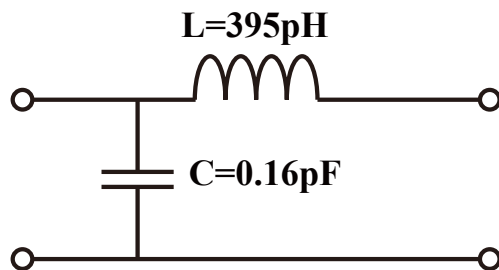


図 40 高速測定用回路で設計した LC 共振回路のパラメータ

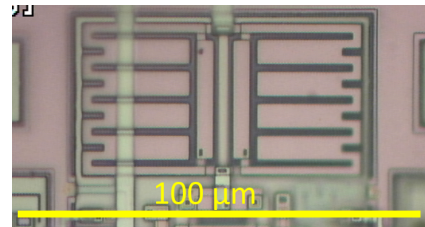


図 41 高速測定用回路で設計した LC 共振回路の顕微鏡写真

実際の設計では LC 共振回路部分の GP 層は取り除いている。インダクタの値が大きいため 2 層を使って設計しているが、その 2 層間のキャパシタの値を共振回路の C として使用している。こうすることによって寄生容量の影響が少なくなり、動作マージンの向上が期待できるためである [22]。このパラメータの場合のインピーダンス Z と遅延時間 τ 、カットオフ周波数 f_C は、 $Z = 50.24\Omega$ 、 $\tau = 7.86\text{ps}$ 、 $f_C = 20.24\text{GHz}$ となる。遅延時間が入力側 JTL よりも短くなってしまったが、これは LC 共振回路の

面積の都合によるものである。後述する数値計算結果もこのパラメータで計算を行っている。

第 IV 章 測定方法

13 出力振幅の測定方法 (低速測定)

回路の低速測定では図 42 に示すような測定系で測定を行った。

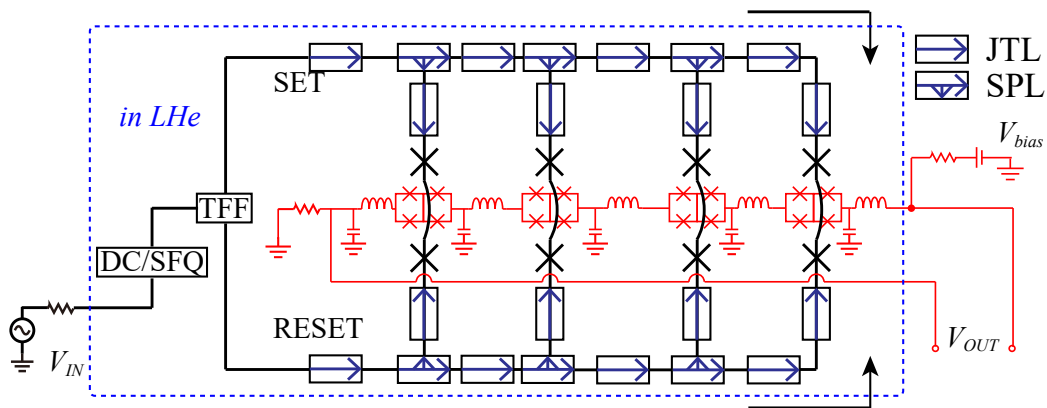


図 42 二重 SQUID 構造分布型アンプの測定系

青枠線の中が液体ヘリウム内である。図には回路の出力からバイアス電流を入力しているが、これ以外にもセルライブラリ用のバイアス電流も別途入力している。回路の出力は増幅部分の両端を低雑音アンプで 100 倍に増幅し、その信号をオシロスコープに入力することで測定している。

低速測定ではバイアス電流 (I_{bias}) に対する出力電圧 (V_{OUT}) を測定する。出力電圧は一定周期毎に SET 状態と RESET 状態が入れ替わった結果となる。

14 出力振幅の測定方法 (高速測定)

作製回路の高速測定は図 43 に示すような測定系で測定した。

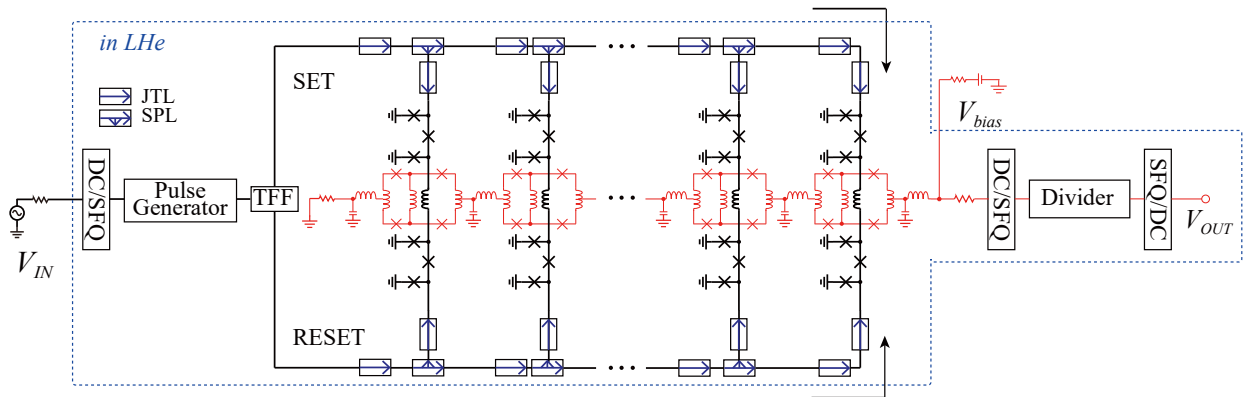


図 43 二重 SQUID 構造分布型アンプの測定系

13 での説明と同様に青点線内は液体ヘリウム環境下である。また図 44 に分周回路の等価回路図を、図 45 に 1GHz の SFQ パルス生成回路の等価回路図を示す。

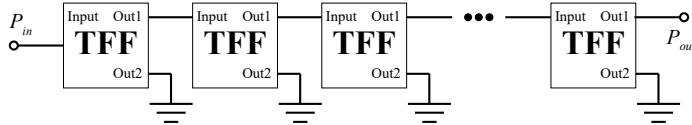


図 44 分周回路の模式図

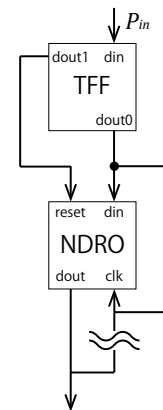


図 45 SFQ パルス
生成回路の模式図

分周回路は TFF の片方の出力のみを次段の TFF に繋げることによって TFF を通る度に SFQ パルスの周期が $1/2$ になることを利用している。今回の設計では 11 段直列接続したため、分周回路で $1/2^{11} = 1/2048$ に分周される。それ以外の部分でも分布型アンプの出力で $1/2$ 、SFQ/dc セルで $1/2$ に分周されるため、回路全体としては $1\text{GHz} / 2^{13} = 122\text{kHz}$ 周期で出力される。SFQ パルス生成回路は NDRO セルの出力をクロックに入力することでその間の遅延時間の周期で SFQ パルス列を生成する回路である。今回は 1GHz の SFQ パルス列を生成するので、遅延時間は 1ns となるように設計している。また、NDRO に入力する SFQ もその前に TFF を通すことで、FG からの入力周期に対して SFQ パル

ス列の生成と停止を繰り返すように設計した。FG からは 2kHz の信号を入力しているため、500us 周期で SFQ パルス列の生成が行われる。

第 V 章

測定結果及び考察

15 数値計算

数値計算では 4 段、12 段、24 段の 3 つについて、単一 SQUID 構造分布型アンプと二重 SQUID 構造分布型アンプの両方についてバイアス電流 (I_{bias}) に対する出力電圧特性と入力周波数に対する出力電圧振幅特性を計算、比較を行った。

15.1 最大出力振幅

15.1.1 4 段分布型アンプ

4 段分布型アンプの電流バイアス対両状態の出力電圧グラフを図 46 に、電流バイアス対出力電圧振幅グラフを図 47 に示す。数値計算では SFQ の入力周波数を 1 Gbps で入力している。

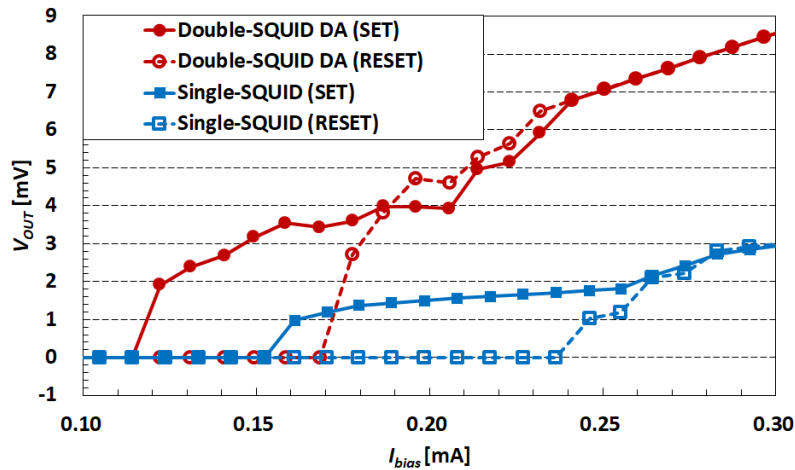


図 46 4 段二重 SQUID 構造分布型アンプの電流バイアス対両状態の出力電圧グラフ

グラフの横軸がバイアス電流、縦軸が両状態の出力電圧、出力電圧振幅であり、それぞれ単位は mA と mV である。また、グラフの赤丸線は二重 SQUID 構造分布型アンプ、青四角線は単一 SQUID 構造分布型アンプの結果を示している。

4 段二重 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.16\text{mA}$ で 3.54mV 、4 段単一 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.24\text{mA}$ で 1.71mV であった。これより、4 段二重 SQUID 構造分布型アンプの最大出力振幅は単一 SQUID 構造分布型アンプのその 2.07 倍となった。

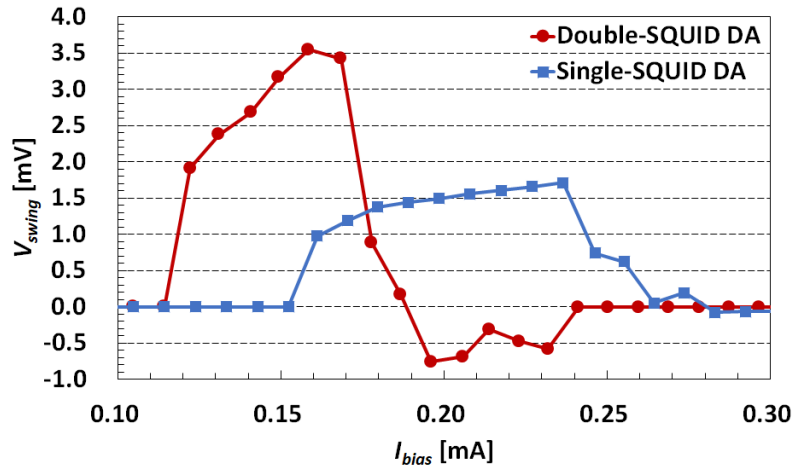


図 47 4 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅グラフ

図 47 において、二重 SQUID 構造分布型アンプに印加するバイアス電流が 0.19mA~0.24 mA の間において負の出力電圧振幅が生じている。 V_{swing} は $V_{swing} = V_{OUT,SET} - V_{OUT,RESET}$ と定義しているが、負の振幅となる領域は $V_{OUT,RESET}$ の電位が $V_{OUT,SET}$ の電位よりも高くなっていることが要因である。このことは図 46 から確認することが可能である。この現象は単一 SQUID 構造分布型アンプでも見られる。

また、バイアス電流に対する動作領域が二重 SQUID 構造の回路と単一 SQUID 構造の回路で異なるが、これは設計段階での SQUID の臨界電流値の差によるものである。

15.1.2 12 段分布型アンプ

12 段分布型アンプのバイアス電流対出力電圧、出力電圧振幅特性を図 48 と図 49 に示す。12 段、24 段では入力周波数 250MHz で SFQ を入力している。

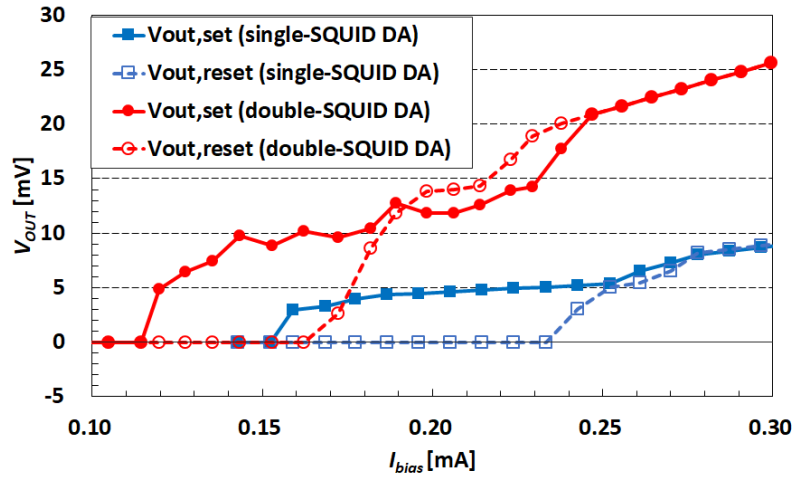


図 48 12 段二重 SQUID 構造分布型アンプの電流バイアス対両状態の出力電圧グラフ

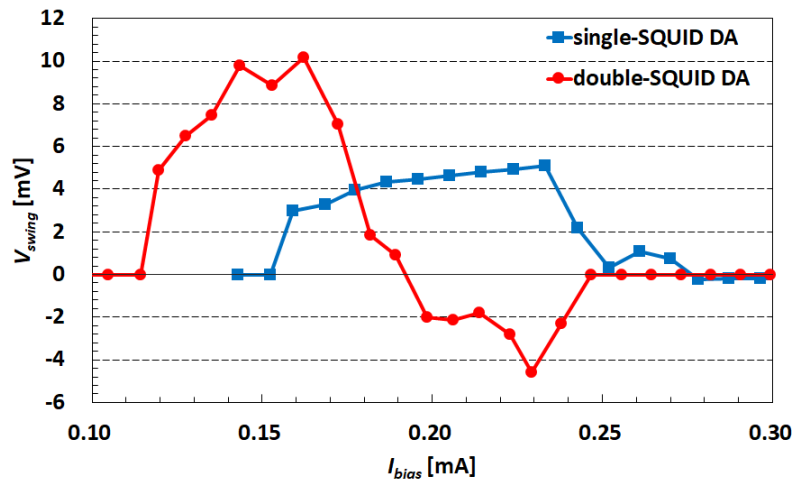


図 49 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅グラフ

12 段二重 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.16\text{mA}$ で 10.15mV 、単一 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.23\text{mA}$ で 5.07mV となった。これより、二重 SQUID 構造分布型アンプの最大出力振幅は 2.00 倍となった。

この結果は室温回路とのインピーダンス整合を考慮していない場合の結果であるが、インピーダンス整合を考慮した場合の出力振幅特性も図 50 と図 51 に示す。

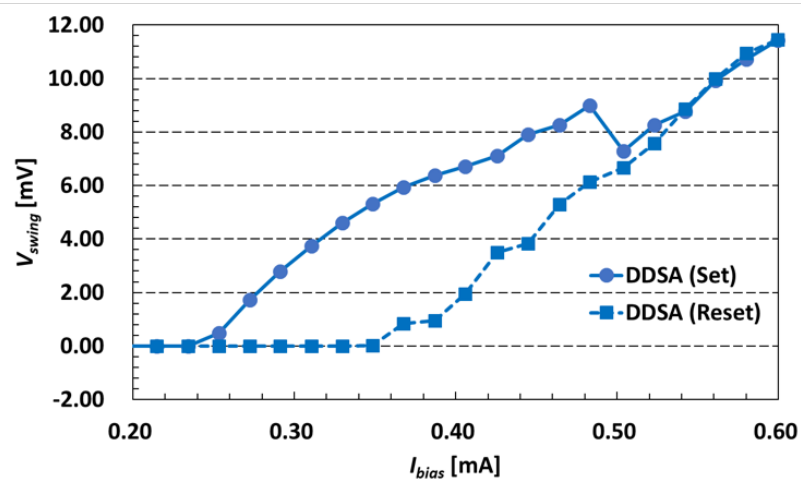


図 50 12 段二重 SQUID 構造布型アンプの電流バイアス対両状態の出力電圧グラフ

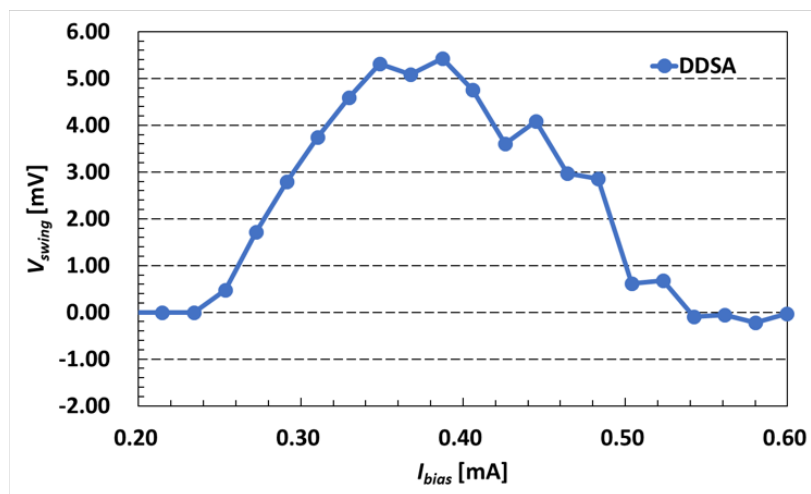


図 51 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅グラフ

整合を取った回路の数値計算結果ではバイアス電流が 0.39mA の時に最大出力振幅が 5.42mV であった。整合を取らない場合と比較して出力電圧振幅は 0.53 倍に減少したが、これは整合を取ることで出力電圧の半分が負荷として消費されるためである。また整合を取ったことにより負荷曲線が変化したため、出力特性は傾いたような特性を取る。整合の有無による負荷曲線の変化を図 52 に示す。

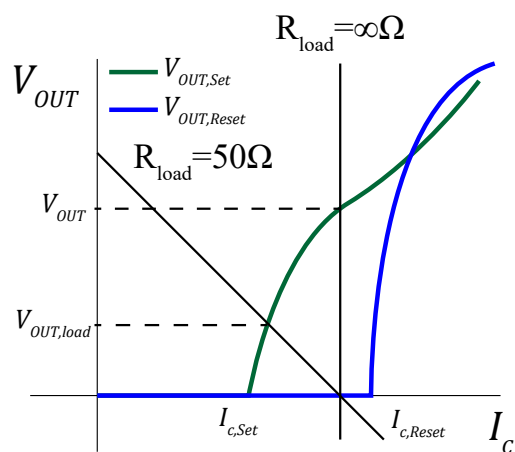


図 52 整合の有無による負荷曲線の変化。SQUID の IV 特性上にプロットしており、 R_{load} で引かれた線と両特性との交点の差が実際に得られる電圧になる。

15.1.3 24 段分布型アンプ

24 段分布型アンプのバイアス電流対出力電圧、出力電圧振幅特性を図 53 と図 54 に示す。

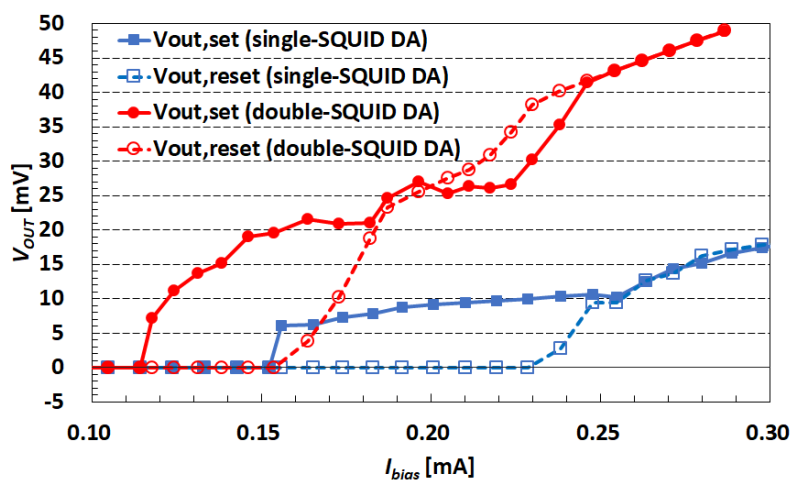


図 53 24 段二重 SQUID 構造分布型アンプの電流バイアス対両状態の出力電圧グラフ。

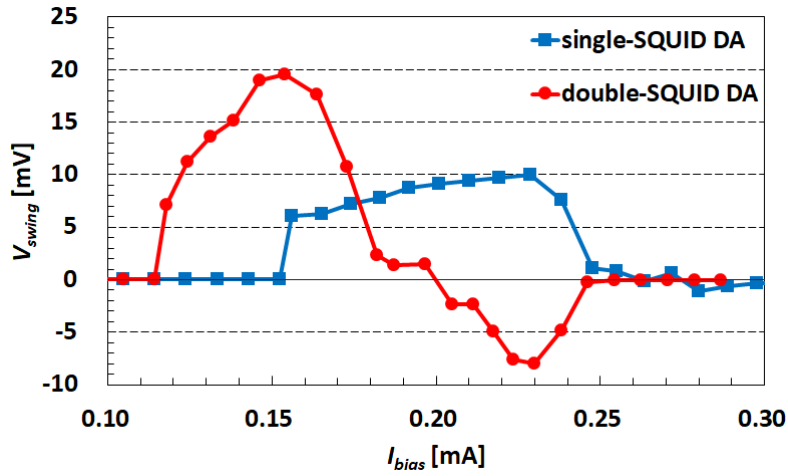


図 54 24 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅グラフ。

24 段二重 SQUID 構造分布型アンプの最大出力電圧振幅は $I_{bias} = 0.15\text{mA}$ で 19.55mV 、単一 SQUID 構造分布型アンプは $I_{bias} = 0.23\text{mA}$ で 10.00mV であった。これより、二重 SQUID 構造分布型アンプの最大出力電圧振幅は単一 SQUID 構造分布型アンプの 1.96 倍となった。

15.2 バイアス電流に対する動作領域

バイアス電流に対する動作領域において、動作とみなす基準電圧を表 5 に示す。

表 5 各分布型アンプ、各段における動作とみなす基準電圧の一覧

構造	4 段 [mV]	12 段 [mV]	24 段 [mV]
single-SQUID	1.2	3.6	7.2
double-SQUID	2.4	7.2	14.4

ここで単一 SQUID 構造 (single-SQUID) では 1 段あたり 0.3mV 、二重 SQUID 構造では 1 段あたり 0.6mV を出力すると仮定した。

表 5 を基に、各分布型アンプの各段におけるバイアス電流に対する動作領域を表 6 に示す。

表 6 各分布型アンプ、各段のバイアス電流に対する動作領域

構造	4 段 [%]	12 段 [%]	24 段 [%]
single-SQUID	± 10.48	± 15.42	± 14.55
double-SQUID	± 7.47	± 6.29	± 6.63

表で示している値は基準電圧以上の領域の中心からの値を示している。各段共に二重 SQUID 構造分布型アンプの動作領域は単一 SQUID 構造分布型アンプよりも狭くなった。狭くなった原因として、二重 SQUID 構造によるものが考えられる。二重 SQUID 構造は等価回路的には 2 つの単一 SQUID が連続して重なった形で示される。この時、2 つの連続した SQUID のインダクタンスは各々の SQUID のパラメータが同じと仮定すると同じ値を取るため、実効的には 2 倍のインダクタンスが存在することになる。 L が大きくなると、SQUID の接合の臨界電流値とインダクタンスの積である LI_C 積が上昇する。この時の臨界電流値の変動量は式 (37) で示される。

$$\Delta I = \frac{2I_0}{1 + \beta} \quad (36)$$

$$\beta = \frac{2LI_0}{\Phi_0} \quad (37)$$

この式より、 L が上昇することによって β が上昇し、外部磁場による臨界電流値の変動量が減少することが分かる。

15.3 入力周波数に対する出力振幅特性

入力周波数 f_{IN} を変化させた時の出力振幅特性も同様に計算を行なった。アンプの動作の基準は得られた出力振幅が最大出力振幅との比で 3dB 以内である場合に動作しているとした。4 段二重 SQUID 構造分布型アンプの最大出力振幅は 3.54mV なので、3dB 減少すると 2.50mV となる。従って周波数を変化させた時の出力振幅が 2.50mV 以上であれば動作していると考ええる。

図 55 に 4 段二重 SQUID 構造分布型アンプの入力周波数に対する出力振幅特性を示す。

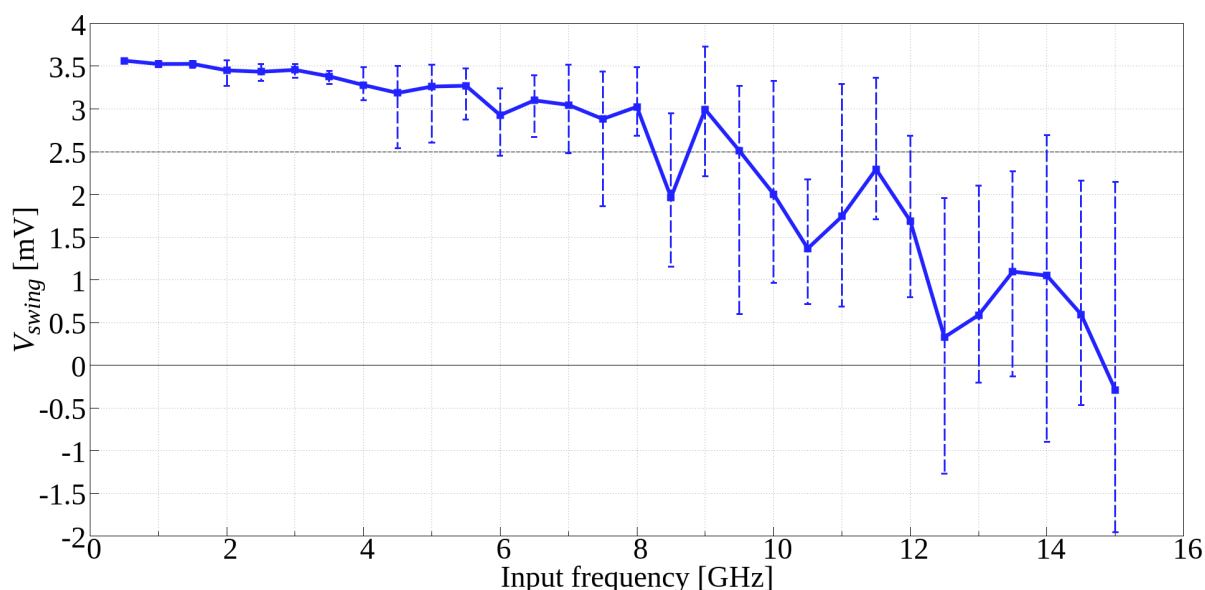


図 55 4 段二重 SQUID 構造分布型アンプの $f_{IN} - V_{swing}$ 特性

横軸が入力周波数、縦軸が出力振幅である。計算時間は 3.0ns 秒間計算を行い、得られた複数の出力振幅の最大値と最小値をエラーバーの範囲としてグラフに表示している。バイアス電流は 4 段回路で最大出力振幅が得られた 0.16mA で固定している。グラフより、 $f_{IN} = 9.5\text{Gbps}$ までは 2.50mV 以上であることから最大動作周波数を 9.5Gbps と言える。途中、8Gbps で出力電圧が落ち込んでいるが、その後の 9Gbps と 9.5Gbps の入力では基準以上の電圧が得られていることから 9.5Gbps が最大動作周波数と判断した。先行研究では 12 段単一 SQUID 構造分布型アンプで 10Gbps での動作を確認しており、先行研究よりも最大動作周波数が低いという結果になった。本研究では出力電圧の増加を念頭に置いて設計を行ったため、高速性は二の次という状態になっているのが原因である。より高速な動作のためには SFQ の伝搬時間とアンプ間の共振回路の伝搬時間との整合を取りつつこの時間を減少させていく必要がある。また出力電圧の立ち上がり、立ち下がり時間も大きく影響するため、なるべくこの影響も少なくする必要がある。これらを基にしたパラメータの改良によってより高速に動作する可能性は十分考えられる。

また、最大出力電圧が得られたバイアス電流の上下 2 点に関しても同様に特性を計算した。その結果を図 56 に示す。

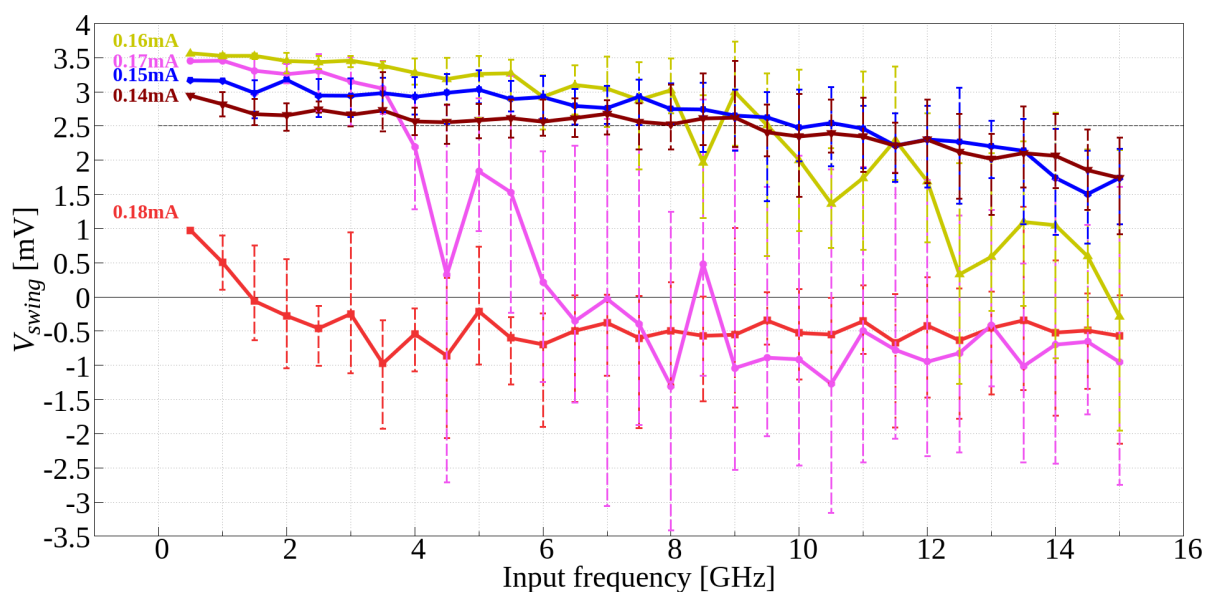


図 56 4 段二重 SQUID 構造分布型アンプの $f_{IN} - V_{swing}$ 特性

グラフの始点の左隣に対応するバイアス電流値を示している。0.16mA よりバイアス電流が大きくなると急激に動作周波数が減少し、0.16mA 以下では概ね近い最大動作周波数が得られる結果になった。

16 低速測定結果

16.1 測定結果と計算結果の比較

16.1.1 4 段二重 SQUID 構造分布型アンプ

4 段二重 SQUID 構造分布型アンプの低速測定結果と数値計算結果を比較したグラフを図 57 と図 58 に示す。両方の図において、赤丸線が測定結果、青三角線が数値計算結果を示している。また、図 57 の実線と破線はそれぞれ SET 状態と RESET 状態の特性を示している。

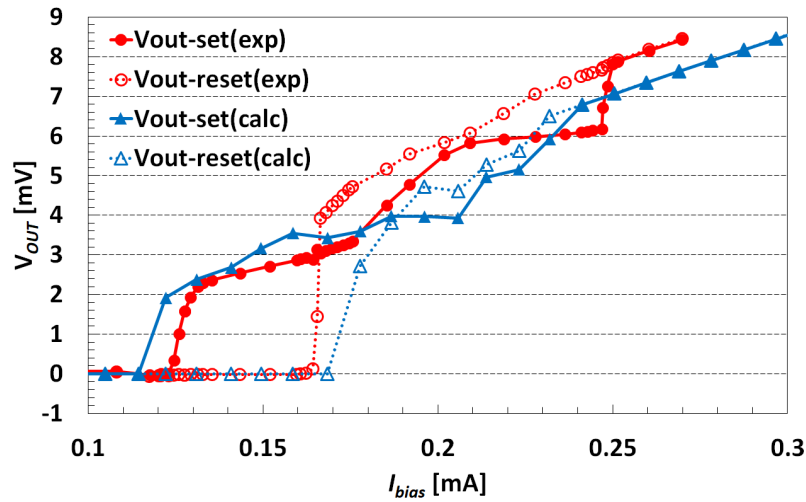


図 57 4 段二重 SQUID 構造分布型アンプの電流バイアス対両状態の出力電圧特性

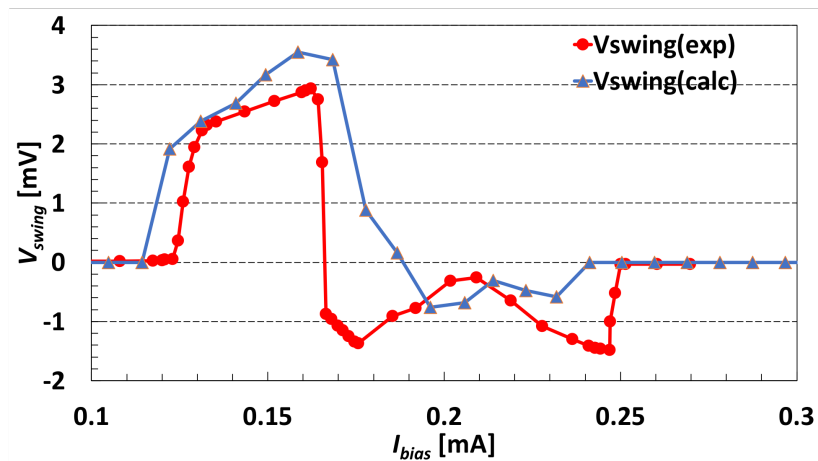


図 58 4 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅特性

図 57 と図 58 はそれぞれバイアス電流に対する出力電圧、出力電圧振幅グラフであり、横軸がそれぞれバイアス電流、縦軸が出力電圧、出力電圧振幅である。測定では、4 段二重 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.16\mu\text{A}$ のとき 2.93mV となった。また、数値計算のグラフと比較すると若干ズレは見られるがその特性は概ね一致した。出力振幅の減少領域で測定結果と数値計算のズレがあるのは Reset 状態での出力電圧の変動が急峻なために生じており、これは寄生容量などに起因すると考えられる。

16.1.2 12 段二重 SQUID 構造分布型アンプ

図 59 に両状態の出力電圧特性を、図 60 に出力電圧振幅特性を示す。グラフ内のプロットの対応は図 57 と 58 と同様である。

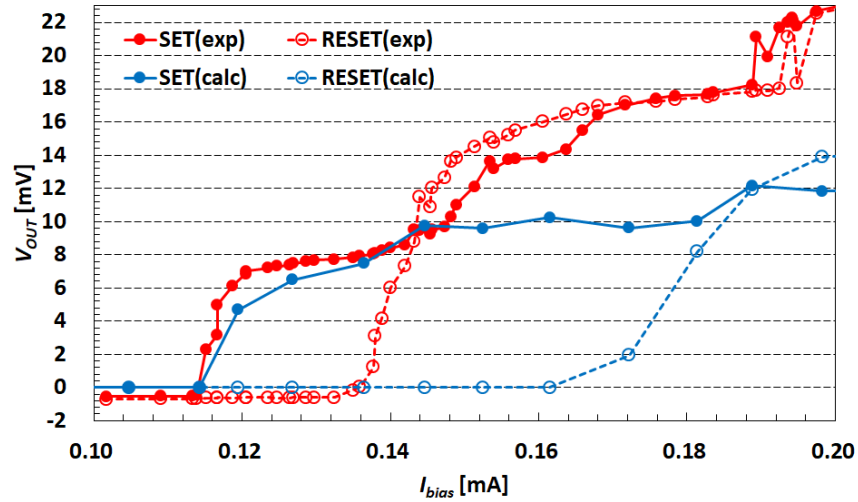


図 59 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧特性

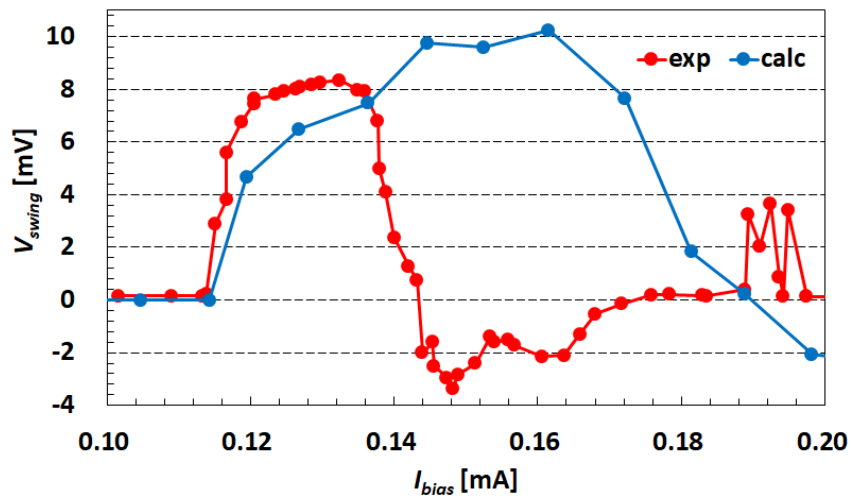


図 60 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅特性

測定結果より最大出力振幅は $I_{bias} = 0.13\mu\text{A}$ で 8.34mV となり、4 段二重 SQUID 構造分布型アン

プの最大出力振幅 2.93mV の 2.85 倍であった。数値計算との比較では最大出力振幅が数値計算より 1.81mV 小さく、バイアス電流に対する動作領域も減少した。正の出力電圧振幅が生じたバイアス電流領域は、数値計算では $69.3\mu\text{A}$ であったのに対して測定結果では $28.1\mu\text{A}$ と約 4 割程度の領域に留まった。図 59 を見ると、SET 状態での特性は数値計算と比較して良好な一致を示しているが、RESET 状態での特性は $130\mu\text{A}$ を境に急激に電圧が上昇しており、動作領域の減少が RESET 特性のズレに起因するといえる。このズレが生じた原因は、外部磁場や残留磁場による影響で RESET 状態での臨界電流値が減少したことが原因として挙げられる。また最大出力電圧が小さくなった要因として、12 段全ての段が同時に動作していないのが原因と考えられる。

12 段二重 SQUID 構造分布型アンプに関しては 2 回の測定で動作領域の異なる出力振幅特性が得られた。図 61 と 62 にその時に得られた特性を示す。

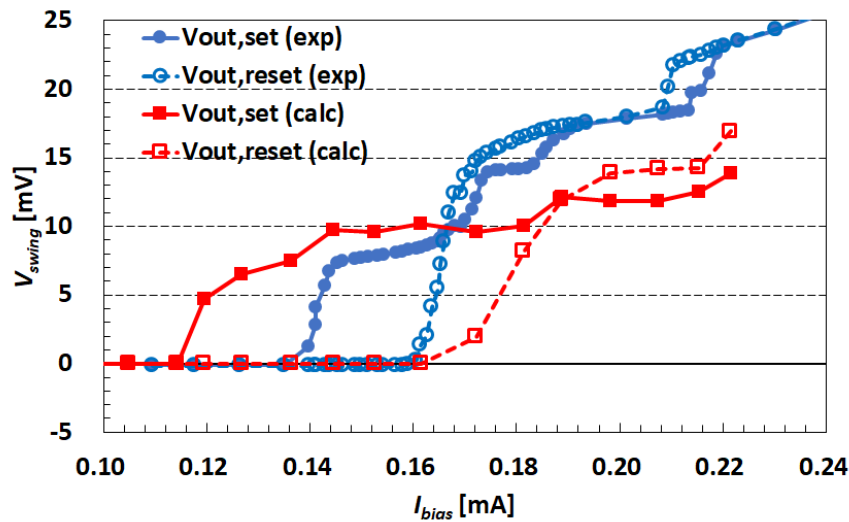


図 61 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧特性

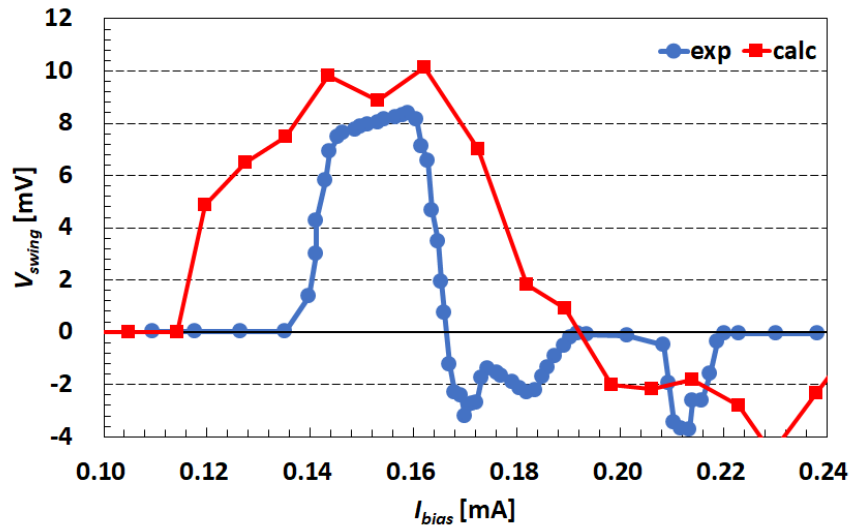


図 62 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅特性

この測定では最大出力振幅は $I_{bias} = 0.16\text{mA}$ で 8.38mV であり、4 段回路の 2.86 倍、数値計算と比較して 1.77mV 小さくなった。この 2 回分の測定結果と数値計算の結果を重ねたグラフを図 63 と 64 に示す。ここで、図 59 と 60 の結果を"exp B"、図 61 と 62 の結果を"exp A"と図中では示している。

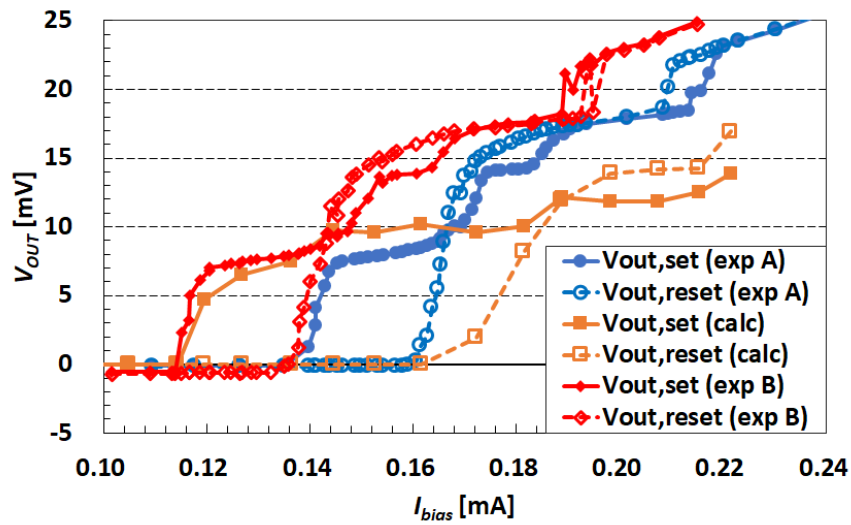


図 63 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧特性

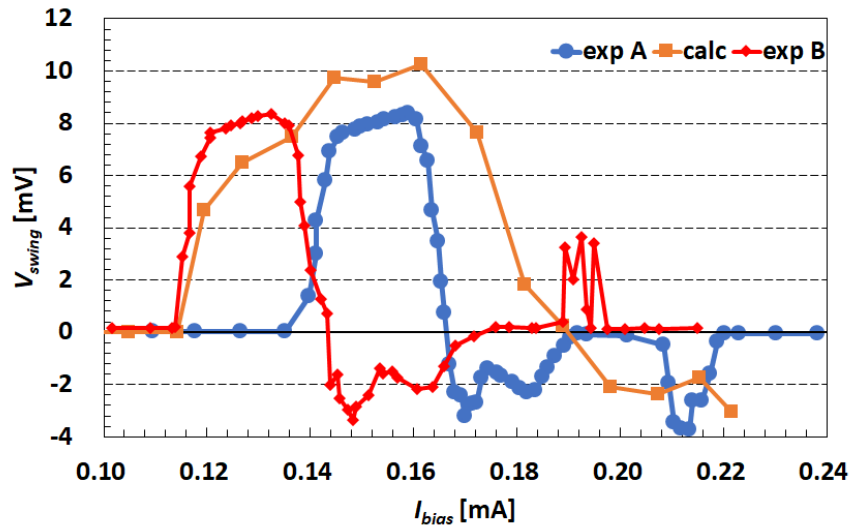


図 64 12 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅特性

また 2 つの測定結果を出力振幅 $V_{swing} > 0.5\text{mV}$ 以上の領域の中心を x 軸の原点に移動した場合の特性を図 65 に示す。

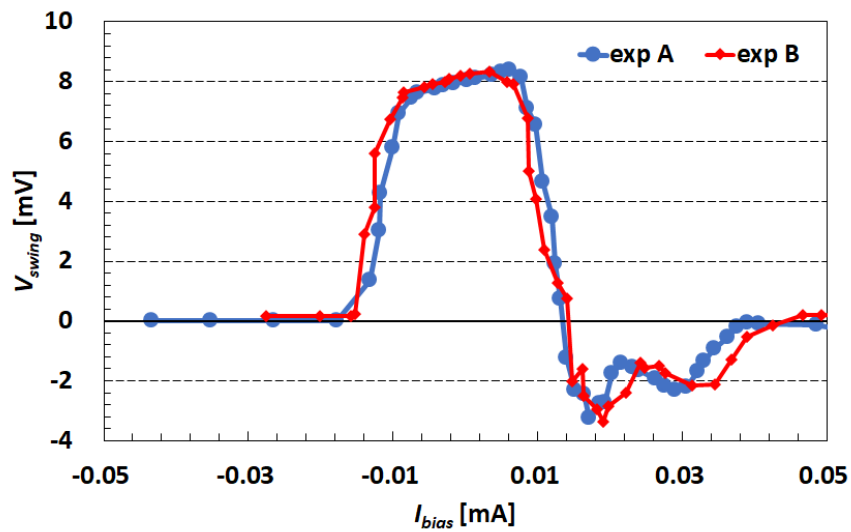


図 65 $V_{swing} > 0.5\text{mV}$ 以上の領域の中心を x 軸原点に移動した場合の出力振幅特性

2 回の測定結果では"exp A"も"exp B"もほぼ同じ動作領域であった。出力振幅の立ち上がりは"exp A"が 0.14mA 、"exp B"が 0.12mA と動作領域 1 つ分ズレで動作し、立ち下がりも同様であった。この動作領域のズレが生じるのは、SQUID の動作点が 2 回の測定で異なっているのが原因として考えられる。

16.1.3 24 段二重 SQUID 構造分布型アンプ

図 66 に両状態の出力電圧特性を、67 に出力電圧振幅特性を示す。グラフ内のプロットの対応は図 57 と 58 と同様である。

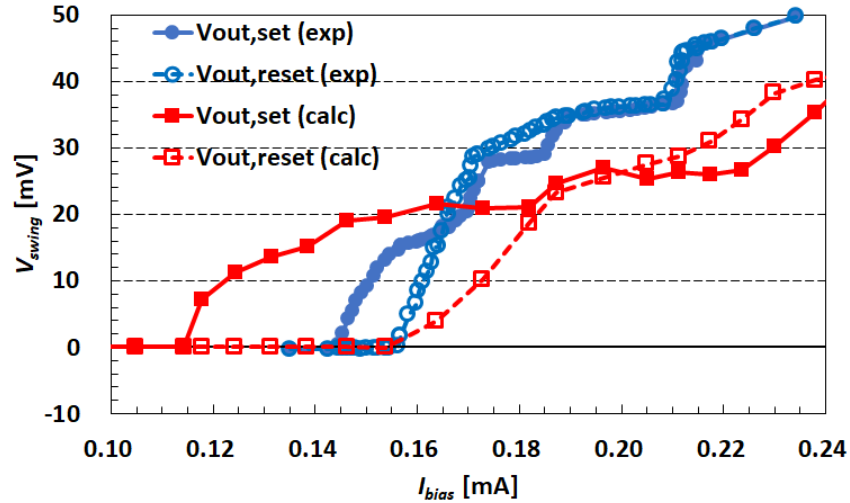


図 66 数値計算と測定結果における 24 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧特性

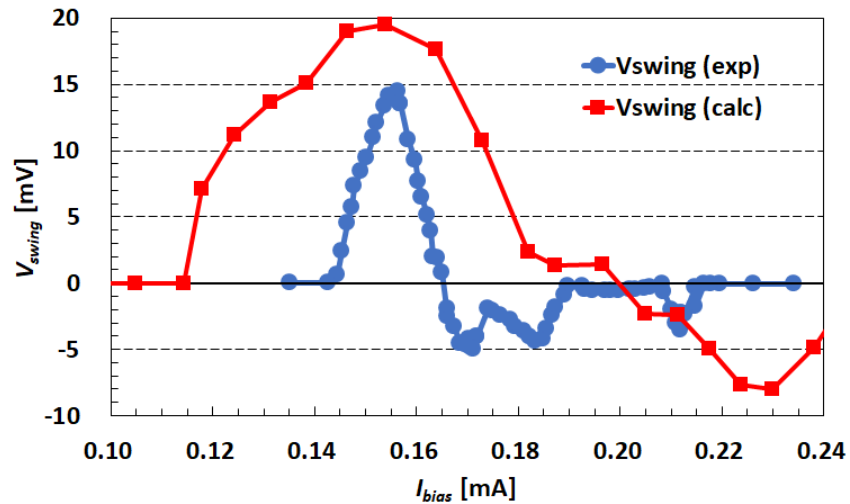


図 67 数値計算と測定結果における 24 段二重 SQUID 構造分布型アンプの電流バイアス対出力電圧振幅特性

測定では 24 段二重 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.16\text{mA}$ で 14.50mV となり、

4 段の場合の最大出力振幅の 4.95 倍、12 段の場合の 1.74 倍となった。数値計算との比較では最大出力振幅が 5.05mV 小さく、動作領域も最大出力振幅の点のみが領域と非常に狭い結果となった。ズレの生じた原因や出力電圧の減少は 12 段の時と同様の原因が考えられる。24 段の場合は RESET 状態の特性は比較的似ているものの、SET 特性の立ち上がりが 0.02mA 後ろにズレていることからリターン電流などの影響で変調を打ち消す方向の電流が流れたと考えられる。

16.2 二重 SQUID 構造分布型アンプと単一 SQUID 構造分布型アンプの測定結果の比較

次に同じプロセスで設計、測定した単一 SQUID 構造分布型アンプの測定結果との比較を行った。4 段の場合のみ設計を行ったため、比較も 4 段のみを行った。図 68 にバイアス電流対出力電圧特性を、図 69 にバイアス電流対出力電圧振幅特性を示す。両特性共に青丸のグラフが二重 SQUID 構造分布型アンプ、赤四角のグラフが単一 SQUID 構造分布型アンプである。また、図 68 のグラフは実線が Set 状態、破線が Reset 状態のグラフである。

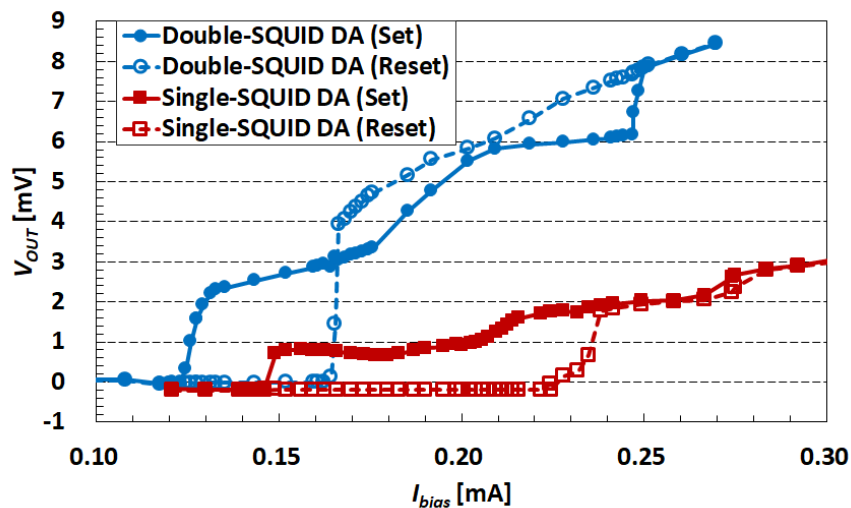


図 68 4 段両分布型アンプの電流バイアス対出力電圧特性

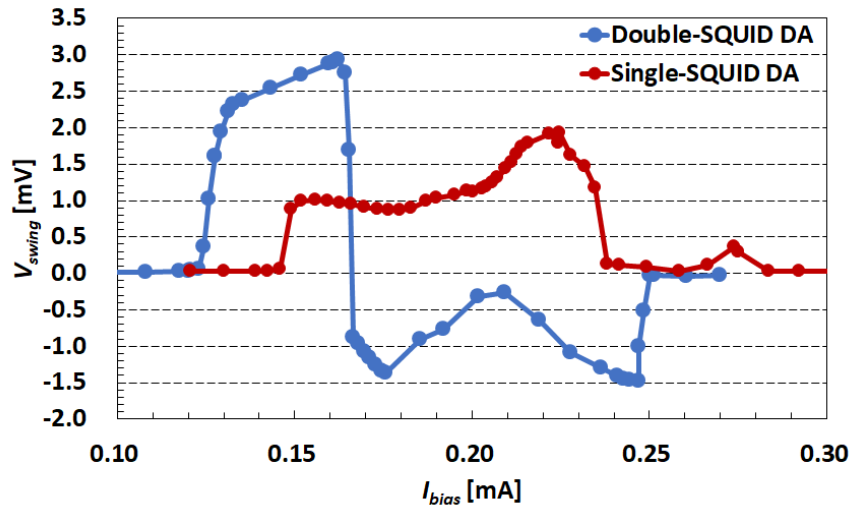


図 69 4 段両分布型アンプの電流バイアス対出力電圧振幅特性

測定で得られた 4 段単一 SQUID 構造分布型アンプの最大出力振幅は $I_{bias} = 0.22\mu\text{A}$ で 1.94mV であった。二重 SQUID 構造分布型アンプの最大出力振幅が 2.93mV なので、最大出力振幅は単一 SQUID 構造分布型アンプの 1.51 倍となった。数値計算では約 2 倍となっていたのが 1.5 倍程度に留まったのは、バイアス電流に対する動作領域の減少の要因でも述べた二重 SQUID 構造が原因であると考えられる。 L が大きくなると、SQUID の接合の臨界電流値とインダクタンスの積である LI_C 積が上昇し、上昇すると接合はスイッチしにくくなる。結果としてスイッチングの速度がその分減少し、その分理想的な 2 倍の出力電圧よりも小さくなったためこのような結果が得られたと考えられる。

バイアス電流に対する動作領域も同様に測定し、比較を行った。出力電圧の基準は表 5 と同様である。表 7 に単一 SQUID 構造分布型アンプと二重 SQUID 構造分布型アンプのバイアス電流に対する動作領域を示す。

表 7 4 段両分布型アンプのバイアス電流に対する動作領域

構造	4 段 [%]
single-SQUID	± 6.10
double-SQUID	± 6.96

表で示している値は基準電圧以上の領域の中心からの値を示している。二重 SQUID 構造分布型アンプの動作領域は単一 SQUID 構造分布型アンプのそれより 1.71% 狭いが、数値計算と比較するとその差は 4.30% 縮まった。差が縮まったのは単一 SQUID 構造分布型アンプの動作領域が数値計算よりも狭い範囲であったのが原因であるが、これは電圧が低い領域では全ての段が同時に動作しなかったためと考

えられる。そのため、動作領域全てで全段同時に動作していた場合はより差が大きくなる。

17 高速測定結果

高速測定では図 43 に示す測定系で 12 段二重 SQUID 構造分布型アンプの測定を行った。図 70 に時間軸に対する出力電圧グラフを示す。

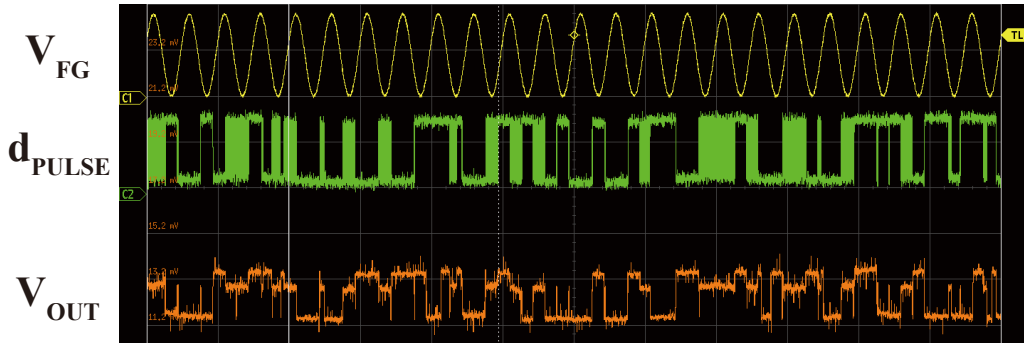


図 70 12 段二重 SQUID 構造分布型アンプの出力電圧の時間依存高速測定結果。

図の横軸は時間軸、縦軸は V_{FG} が FG からの入力波形、 d_{PULSE} が生成された SFQ の確認用回路の出力、 V_{OUT} がアンプの出力である。 d_{PULSE} は 1GHz 周期で生成された SFQ パルス列の確認用回路の出力であり、分周回路を通じて $1/2^{12}$ に分周された波形を出力する。FG の入力周波数は 2kHz なので、0.5ms 毎に 244kHz で発振する波形が見られる。その時の概形を図 71 に示す。

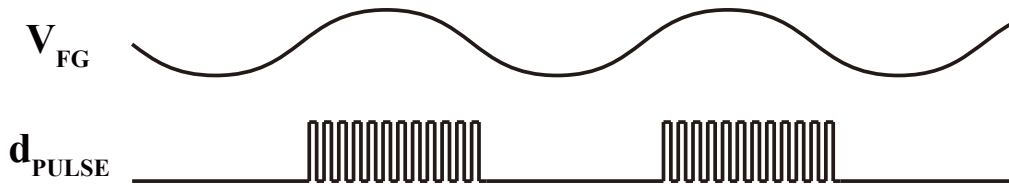


図 71 生成された SFQ を分周回路に通した場合に正常動作していれば見れる波形。入力波形の 2 倍の周期で SFQ が生成されるため、SFQ に対する反応がある場合と無い場合を繰り返す波形となる。

図 70 で見ている波形は図 71 に示す概形とは異なる形となっており、SFQ の生成がうまくいっていないことが確認できる。この要因として挙げられるのは、dc/SFQ セルでの SFQ 生成がうまく出来ていない、NDRO セルのセットとリセットが正常に動作していない、何らかの要因により分周回路で SFQ が消滅してしまっているなどがある。また、発振回路から 1GHz の SFQ パルス列が生成されるが、この周

期が乱れたり SFQ が正常に伝搬しない場合も同様に正常に動作しないことが考えられる。発振周期の乱れは図 70 の d_{PULSE} の発振周期を見ることで 1GHz 周期での SFQ 生成ができていないかの確認が可能である。入力回路に設置した分周回路は分周回路、SFQ/dc セルを通ると $1/2^{12}$ 周期に分割される。1GHz の入力を入れた場合は $1\text{GHz}/2^{12} \simeq 244\text{kHz}$ で振動していれば正常に動作していることが確認できる。図 72 に示すのは図 70 の結果から振動している部分を拡大したグラフである。

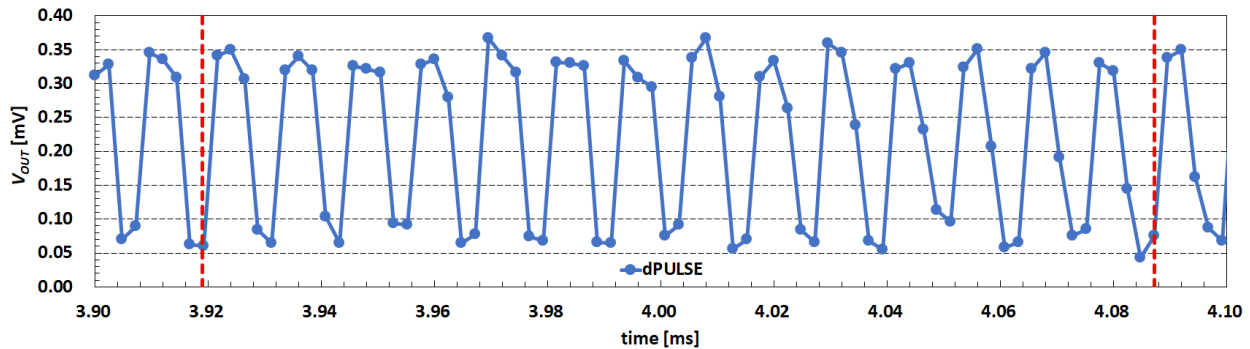


図 72 d_{PULSE} の発振部分を拡大したグラフ。赤色の点線で囲っている部分の間周期の数から 1 周期あたりの平均時間を算出して求める。

横軸は時間の経過を示しており、縦軸は出力電圧を示している。グラフ内に 2 つある赤色の破線の間の時間とその間の周期の数から 1 周期あたりの平均時間を算出することで周波数を逆算する。グラフより破線間の時間は 0.168 ms、周期は 14 周期であることが確認できるため、1 周期あたりの平均時間は $0.168/14 = 0.012\text{ms}$ と求まる。この逆数が周波数なので、振幅の周波数は $1/0.012 = 83.3\text{kHz}$ と求まる。設計上では 244kHz で出力されるように設計しているのに、その 0.34 倍の周波数で出力されていることが分かった。設計よりも大幅に小さな周期で出力された原因としては、バイアス電流値のズレや発振回路の誤動作などが挙げられる。

また V_{OUT} はアンプの出力をそのまま観測しているため、信号は 500MHz でオシロスコープに入力される。しかしオシロスコープ側がそこまで帯域幅が無いため、このようなまばらな波形が表われている。このような測定になったのは、アンプの出力側に設置している分周回路の動作領域が見つからなかったことが原因である。

正常動作を確認することができなかったが、入力波形と同期する形で出力波形も変化しているので、正常ではないが何かしらの程度の動作はしているということが確認できる。

第 VI 章

結論

超伝導出力増幅回路の一種である RSFQ 分布型アンプについて、より高い出力電圧振幅を得るために 1 段あたりの SQUID の数を 1 個から 2 個に増やした二重 SQUID 構造分布型アンプを設計し、出力電圧振幅の向上を図った。各段の SQUID の数を 2 個に増やすことによって 1 段あたりの出力電圧振幅を 2 倍にすることが期待されるためである。

数値計算においては、始めに動作検討として 4 段二重 SQUID 構造分布型アンプと 4 段単一 SQUID 構造分布型アンプを設計し、両回路のバイアス電流に対する出力電圧特性を計算し、同時にバイアス電流に対する正の出力電圧振幅が生じる動作領域も同時に計算、比較を行なった。その後、12 段と 24 段についても同様の比較を行った。

4 段二重 SQUID 構造分布型アンプでは最大出力振幅が 3.54mV、単一 SQUID 構造分布型アンプでは 1.71mV となり、二重 SQUID 構造分布型アンプのは単一 SQUID 構造のそれと比較して 2.07 倍に最大出力振幅が増加した。一方、バイアス電流に対する動作領域は二重 SQUID 構造分布型アンプで $\pm 7.47\%$ 、単一 SQUID 構造分布型アンプでは $\pm 10.48\%$ となり、二重 SQUID 構造分布型アンプの動作領域は単一 SQUID 構造のそれと比較して 6.02% 小さくなった。これは SQUID を 2 個に増やしたために接合を流れる環状電流が減少したのが原因であると考えられる。

12 段二重 SQUID 構造分布型アンプでは最大出力振幅が 10.15mV となり、単一 SQUID 構造の場合の 5.07mV と比較して 2.00 倍最大出力電圧が増加した。バイアス電流に対する動作領域については二重 SQUID 構造分布型アンプで $\pm 6.29\%$ 、単一 SQUID 構造の場合で $\pm 15.42\%$ となり、18.26% 減少した。

24 段二重 SQUID 構造分布型アンプでは最大出力振幅が 19.55mV、単一 SQUID 構造で得られた 10.00mV と比較して 1.96 倍最大出力電圧が増加した。バイアス電流に対する動作領域については二重 SQUID 構造分布型アンプで $\pm 6.63\%$ 、単一 SQUID 構造の場合で $\pm 14.55\%$ となり、15.84% 減少した。

測定では 4 段、12、24 段二重 SQUID 構造分布型アンプで 1kHz の低速測定を、12 段二重 SQUID 構造分布型アンプで 1GHz の高速測定を実施した。4 段では単一 SQUID 構造分布型アンプとの比較も実施した。4 段二重 SQUID 構造分布型アンプの低速測定では、最大出力電圧振幅が 2.94mV となった。またバイアス電流に対する出力振幅特性も数値計算の概形と概ね一致するなど、良好な結果が得られた。

単一 SQUID 構造の回路との比較では単一 SQUID 構造の回路の最大出力振幅が 1.94mV であったことから、二重 SQUID 構造分布型アンプの最大出力振幅は単一 SQUID 構造の場合の 1.51 倍となった。数値計算とは異なり 2 倍を下回る結果となったが、これも SQUID の数が 2 個に増えたのが要因として考えられる。等価回路的には直列に SQUID が 2 個繋がっていることになるので、実質的なインダクタンスが上昇する。これにより SQUID の LI_C 積が増加するため、SQUID の接合はスイッチしにくくなり、出力電圧が小さくなったと考えられる。また、バイアス電流に対する動作領域では単一 SQUID 構造

分布型アンプで 13.91%、二重 SQUID 構造分布型アンプの場合で 12.20% と比較的近い動作領域となった。これは単一 SQUID 構造分布型アンプの出力振幅電圧がバイアス電流が小さい領域において設計よりも小さい電圧しか出なかったために領域が小さくなったことが分かった。設計よりも小さい電圧が出る要因としては、SQUID の臨界電流値やパラメータのズレにより各段が同時に動作しなかったために生じたと考えられる。

12 段二重 SQUID 構造分布型アンプでは低速測定と高速測定の両方を行なった。低速測定では最大出力振幅が 8.34mV となったが、設計上は最大 10.2mV 出るはずであり、2mV 程度出力電圧が減少した。また、バイアス電流に対する動作領域も設計と比較して 4 割程度の領域に留まった。SET と RESET 両方の状態を数値計算と比較すると、RESET 状態の方で臨界電流値に大きな差があるためにこのような結果になったことが分かった。臨界電流値が大きく減少した理由としては、外部磁場や回路を動作させることによる残留磁場の影響を受けて臨界電流値が減少したと考えられる。

24 段二重 SQUID 構造分布型アンプの低速測定では最大出力振幅が 14.50mV であった。数値計算と比較すると 5.05mV 最大出力振幅が減少し、バイアス電流領域も最大出力振幅の 1 点のみであった。ズレや出力電圧の減少の原因は 12 段の回路と同様の原因が考えられる。

12 段二重 SQUID 構造分布型アンプの高速測定では回路に 1GHz の SFQ パルス列を入力した。測定では、入力 SFQ に対して何らかの反応は見られたものの、回路の動作には至らなかった。入力 SFQ を確認すると、FG からの入力に対して正常に動作していなかったため、周期的に SFQ パルス列が生成できていないことが分かった。この現象が生じるのは、dc/SFQ が FG からの入力に対して正常に SFQ を生成できていない、NDRO セルによる SFQ パルス列の生成が正常に出来ていない、分周回路の誤作動によるものなどの要因が考えられるが、この実験結果だけでは原因を特定できなかった。また、SFQ パルス列を分周回路に通した測定結果では、発振している領域において出力 SFQ の周期がどれぐらいかを確認した。正常に動作していれば 244kHz 周期で出力があるが、実際に調べたところ 83kHz であり約 1/3 倍の周波数で出力されていた。このことより、SFQ パルス列の生成においても何かしらの問題があるということが判明した。パルス列の生成においては、発振回路の誤動作やバイアス電流値が動作領域と異なってしまうなどの要因が考えられる。

高速測定においては満足のいく結果を得られなかったが、低速測定においては二重 SQUID 構造分布型アンプの動作を確認することが出来た。今後高速測定でも動作するように分布型アンプを改良することで超伝導回路と室温回路とのデータリンクが可能になると考えられる。

参考文献

- [1] 岸野正剛. 超伝導エレクトロニクスの物理. 丸善株式会社, Sep. 1993.
- [2] K. Nakajima, H. Mizusawa, H. Sugahara, and Y. Sawada. Phase mode josephson computer system. *IEEE Transactions on Applied Superconductivity*, Vol. 1, No. 1, pp. 29–36, March 1991.
- [3] K. Nakajima, Y. Onodera, and Y. Ogawa. Logic design of josephson network. *Journal of Applied Physics*, Vol. 47, No. 4, pp. 1620–1627, 1976.
- [4] K. K. Likharev and V. K. Semenov. Rsfq logic/memory family: a new josephson-junction technology for sub-terahertz-clock-frequency digital systems. *IEEE Transactions on Applied Superconductivity*, Vol. 1, No. 1, pp. 3–28, March 1991.
- [5] W. Chen, A. V. Rylyakov, V. Patel, J. E. Lukens, and K. K. Likharev. Rapid single flux quantum t-flip flop operating up to 770 ghz. *IEEE Trans. Appl. Supercond.*, Vol. 9, No. 2, pp. 3212–3215, June 1999.
- [6] H. A. Wheeler. Wide-band amplifiers for television. *Proceedings of the IRE*, Vol. 27, No. 7, pp. 429–438, July 1939.
- [7] E. L. Ginzton, W. R. Hewlett, J. H. Jasberg, and J. D. Noe. Distributed amplification. *Proceedings of the IRE*, Vol. 36, No. 8, pp. 956–969, Aug 1948.
- [8] D.M. Pozar. *Microwave Engineering, 4th Edition*. Wiley, 2011.
- [9] Quentin P. Herr. Stacked double-flux-quantum output amplifier. In *IEEE Transactions on Applied Superconductivity*, 2005.
- [10] Michael Tinkham. 超伝導現象. 産業図書株式会社, 初版, Sep. 1981.
- [11] 丸山道隆. ジョセフソン電圧標準の現状. *AIST Bulletin of Metrology*, Vol. 8, No. 2, 2011.
- [12] 小谷誠. Squid による生体磁気計測. 可視化情報学会誌, Vol. 14, No. 55, pp. 214–218, 1994.
- [13] 中島康治. 単一磁束量子を用いた電子素子と回路. 応用物理, Vol. 62, No. 5, pp. 467–470, 1993.
- [14] 寺井弘高, 山下太郎, 三木茂人, 牧瀬圭正, 王鎮. 超伝導単一磁束量子 (sfq) 論理回路. 低温工学, Vol. 48, No. 2, pp. 77–82, 2013.
- [15] Takehiko TANABE. セシウム一次周波数標準の現状とその高精度化に向けた最近の研究動向. Aist bulletin of metrology, National Institute of Advanced Industrial Science and Technology, 2013.
- [16] 早川尚夫, 菅野卓雄. 超高速デジタルデバイス・シリーズ 4 超高速ジョセフソン・デバイス. 株式会社 培風館, Nov. 1986.
- [17] 吉田二郎. 単一磁束量子素子を用いた論理回路技術. 応用物理, Vol. 67, No. 4, pp. 410–416, 1998.
- [18] H. Suzuki, T. Imamura, and S. Hasuo. Applications of synchronized switching in series-parallel-connected josephson junctions. *IEEE Trans. Electron Dev.*, Vol. 37, No. 11, pp. 2399–2405,

Nov 1990.

- [19] Hiroshi Nakagawa, Eiichi Sogawa, Shin Kosaka, Susumu Takada, and Hisao Hayakawa. Operating characteristics of josephson four-junction logic (4jl) gate. *Japan. J. Appl. Phys.*, Vol. 21, No. 4A, p. L198, 1982.
- [20] A. Hebard, S. Pei, L. Dunkleberger, and T. Fulton. A dc-powered josephson flip-flop. *IEEE Trans. Magn.*, Vol. 15, No. 1, p. 408, January 1979.
- [21] R. P. Welty and J. M. Martinis. A series array of dc squids. *IEEE Transactions on Magnetism*, Vol. 27, No. 2, p. 2924, March 1991.
- [22] Q P Herr. A high-efficiency superconductor distributed amplifier. *Supercond. Sci. & Tech.*, Vol. 23, No. 2, p. 022004, 2010.
- [23] Theodore Van Duzer, Lizhen Zheng, Stephen R. Whiteley, Hoki Kim, Jaewoo Kim, Xiaofan Meng, and Thomas Ortlepp. 64-kb hybrid Josephson-CMOS 4 Kelvin RAM with 400 ps access time and 12 mW read power. *IEEE Trans. Appl. Supercond.*, Vol. 23, No. 3, pp. 8–11, 2013.
- [24] Fumihiro China, Naoki Takeuchi, Thomas Ortlepp, Yuki Yamanashi, and Nobuyuki Yoshikawa. [invited talk] proposal of parallel to serial conversion circuit for 2-dimension superconductive detector array using adiabatic quantum-flux-parametron. *IEICE technical report*, Vol. 117, No. 428, pp. 33–38, jan 2018.
- [25] D. F. Schneider, J. C. Lin, S. V. Polonsky, V. K. Semenov, and C. A. Hamilton. Broadband interfacing of superconducting digital systems to room temperature electronics. *IEEE Trans. Appl. Supercond.*, Vol. 5, No. 2, pp. 3152–3155, June 1995.
- [26] Martin E Huber, Patric A Neil, Robert G Benson, Dehorah A Bums, Alan Corey, Christopher S Flynn, Yevgeniya Kitaygoradskaya, Omid Massihzaueh, John M Martinis, and G C Hilton. DC SQUID Series Array Amplifiers with 120MHz Bandwidth (Corrected). *IEEE Trans. Appl. Supercond.*, Vol. 11, No. 2, pp. 4048–4053, 2001.
- [27] 永沢秀一. Nb 標準プロセス (stp2) の設計仕様, Apr. 2015.
- [28] S. Nagasawa, Y. Hashimoto, H. Numata, S. Tsuchida, and S. Tahara. 380 ps, 9.5 mw josephson 4 kbit ram. *Electronics Letters*, Vol. 30, No. 10, pp. 761–762, May 1994.
- [29] Naoki Takeuchi, Shuichi Nagasawa, Fumihiro China, Takumi Ando, Mutsuo Hidaka, Yuki Yamanashi, and Nobuyuki Yoshikawa. Adiabatic quantum-flux-parametron cell library designed using a 10 ka/cm² niobium fabrication process. *Superconductor Science and Technology*, Vol. 30, p. 035002, 03 2017.
- [30] Shuichi Nagasawa, Kenji Hinode, Tetsuro Satoh, Mutsuo Hidaka, Hiroyuki Akaike, Akira Fujimaki, Nobuyuki Yoshikawa, Kazuyoshi Takagi, and Naofumi Takagi. Nb 9-layer fabrication

- process for superconducting large-scale sfq circuits and its process evaluation. *IEICE Transactions on Electronics*, Vol. 97, No. 3, pp. 132–140, 2014.
- [31] 信行吉川. 単一磁束量子回路を用いた超高速マイクロプロセッサの開発とその展望. 電子情報通信学会論文誌. C, エレクトロニクス, Vol. J91-C, No. 3, pp. 183–193, mar 2008.
 - [32] S Yorozu, Y Kameda, H Terai, A Fujimaki, T Yamada, and S Tahara. A single flux quantum standard logic cell library. *Physica C*, Vol. 378-381, pp. 1471 – 1474, 2002.
 - [33] E. S. FANG. A josephson integrated circuit simulator (jsim) for superconductive electronics application. *Extended Abstracts of 1989 Int. Supercond. Electronics Conf. (The Japan Society of Appl. Phys., Tokyo, 1989)*, 1989.
 - [34] N Mori, A Akahori, T Sato, N Takeuchi, A Fujimaki, and H Hayakawa. A new optimization procedure for single flux quantum circuits. *Physica C*, Vol. 357-360, pp. 1557 – 1560, 2001.
 - [35] Laurence W. Nagel. *SPICE2: A Computer Program to Simulate Semiconductor Circuits*. PhD thesis, EECS Department, University of California, Berkeley, 1975.
 - [36] Toshimitsu Morooka. Design, fabrication and evaluation of a four-josephson-junction superconducting quantum interference device with two superconducting loops. *Japan. J. Appl. Phys.*, Vol. 36, No. 12A, p. L1587, 1997.
 - [37] C. J. Fourie and W. J. Perold. Simulated inductance variations in rsfq circuit structures. *IEEE Trans. Appl. Supercond.*, Vol. 15, No. 2, pp. 300–303, June 2005.

謝辞

本研究は、電気通信大学 情報理工学研究科 基盤理工学専攻 水柿研究室において行われました。

本研究を進めるのに際し、熱心なご指導を下さいました水柿義直教授に感謝致します。また、研究の進め方や測定方法などに関して様々な助言を下さいました守屋雅隆助教に感謝致します。装置の使用方法等について親切なご指導を下さった島田宏教授に感謝致します。

回路設計や測定方法、シミュレーション等に関して様々なご指導を下さった荒井祐真氏、貴重なご意見や提案をしていただいた SET グループの松本和彦氏に感謝致します。また、同じ超伝導集積回路グループにおいて有用な助言やご指摘を頂きました山崎洸生氏、曾明裕太氏や研究室の同期として様々な助言を頂きました森林誠氏、高島匠吾氏に感謝致します。そして、水柿研究室、島田研究室、小久保研究室の皆様に感謝致します。

本研究の一部は、JSPS 科研費 17K04979 の助成と、東京大学 VDEC を通じて日本ケイデンス株式会社の協力で行われました。本研究に使用された回路は、(独) 産業総合技術研究所 (AIST) の超伝導クリーンルーム (CRAVITY) において、AIST-STP2 プロセスを用いて作製されました。AIST-STP2 は、(公財) 国際超伝導産業技術研究センター (ISTEC) のニオブ集積回路プロセスを AIST にて発展させたものです。また測定には液体ヘリウムを使用しており、その供給は電気通信大学 低温室からの安定的な供給によって実現されました。

発表実績

学会発表

- K. Higuchi, H. Shimada, and Y. Mizugaki "Design and Operation of a Distributed Amplifier Comprising Double-SQUID Elements", The 12th Superconducting SFQ VLSI Workshop (SSV 2019), Kobe, Japan, January 16-17, 2019.
- K. Higuchi, H. Shimada, and Y. Mizugaki, "Design and Operation of Distributed Double-SQUID Amplifier for RSFQ Circuits", The 31st International Symposium on Superconductivity (ISS 2018), Tsukuba, Ibaraki, Japan, December 12-14, 2018.
- 樋口孔明, 島田 宏, 水柿義直, "二重 SQUID 構造 RSFQ 分布型アンプの特性評価", 2018 年電子情報通信学会ソサイエティ大会, 2018 年 9 月 14 日
- 樋口孔明, 島田 宏, 水柿義直, "RSFQ 回路用分布型アンプのバイアス電流-出力電圧特性", 2018 年電子情報通信学会総合大会, 2018 年 3 月 21 日

その他発表実績

- 樋口孔明, 荒井祐真, 水柿義直, "DFQ アンプの誤動作時における磁束量子ダイナミクス", 2017 年度春季 第 94 回 低温工学・超電導学会研究発表会, 2017 年 5 月 23 日

付録 A 単一 SQUID 構造分布型アンプの結果

ここでは、以前作製した 12 段単一 SQUID 構造分布型アンプの結果について載せる。この回路は二重 SQUID 構造分布型アンプの作製の前に設計した回路であり、我々の使用するプロセスにおいて過去に分布型アンプが実証されたことが無かったことから作製した。なおこの結果は"RSFQ 回路用分布型アンプのバイアス電流-出力電圧特性"という題目で 2018 年の電子情報通信学会総合大会で発表した。

A.1 設計した回路のパラメータ

作製したのは 12 段単一 SQUID 構造分布型アンプであり、回路構成などは図 42 と同じである。図 73 に各段のパラメータを記す。

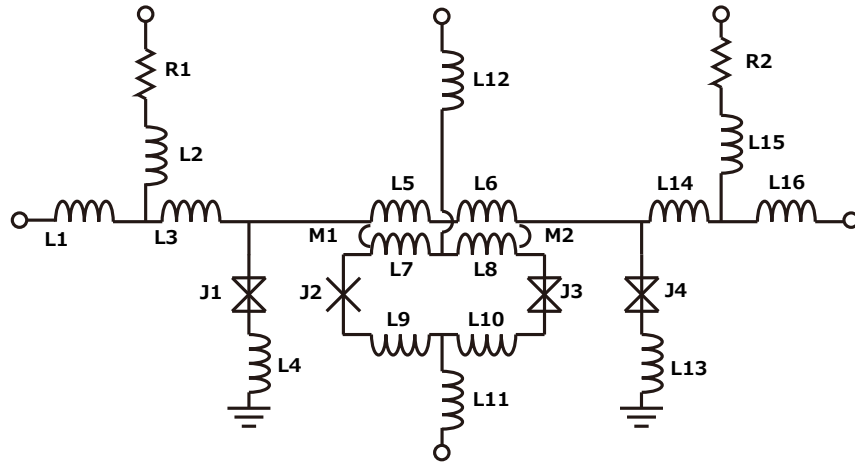


図 73 単一 SQUID 構造分布型アンプの 1 段分の等価回路図。パラメータは JSIM と SCOPE2 を用いて求めた; $L1 = L2 = L3 = L14 = L15 = L16 = 0.50\text{pH}$, $L4 = 0.26\text{ pH}$, $L5 = L6 = 5.70\text{ pH}$, $L7 = L8 = 3.50\text{ pH}$, $L9 = L10 = 0.38\text{ pH}$, $L11 = L12 = 125\text{ pH}$, $L13 = 0.27\text{ pH}$, $J1 = J4 = 0.22\text{ mA}$ ($1.73\text{ }\Omega$), $J2 = 0.14\text{ mA}$, $J3 = 0.14\text{ mA}$ ($2.07\text{ }\Omega$), $R1 = 12.97\Omega$, $R2 = 16.16\Omega$, $M1 = M2 = 3.82\text{pH}$ ($k = 0.428$).

このパラメータを元に設計した回路全体の顕微鏡写真図を 74 に、1 段を拡大した図を 75 に示す。

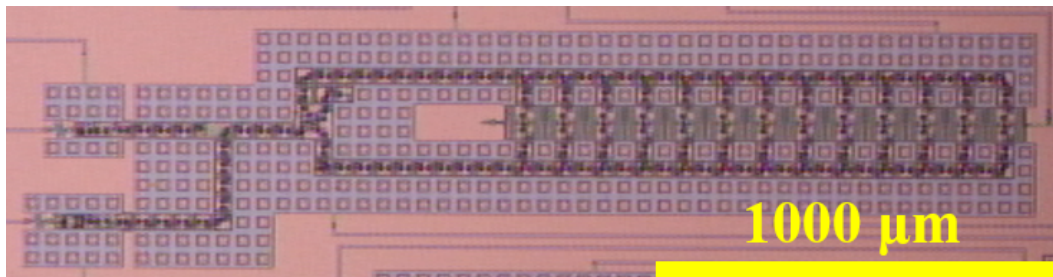


図 74 12 段単一 SQUID 構造分布型アンプ全体の顕微鏡写真図

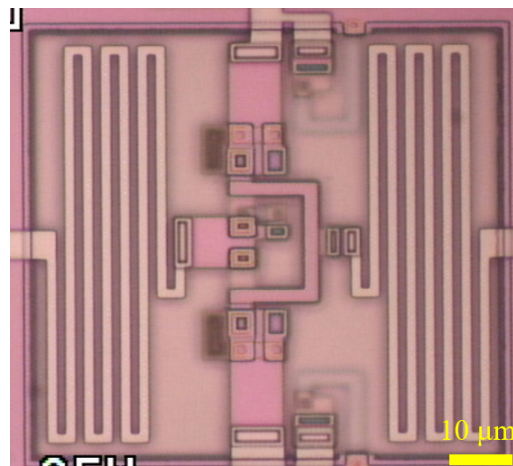


図 75 単一 SQUID 構造分布型アンプ 1 段を拡大した顕微鏡写真図

パラメータの接合の後ろにある抵抗値は接合と並列に接続されたシャント抵抗である。また図中の × 印の記号はアンダーダンプ接合、蝶ネクタイ印の記号はオーバードンプ接合である。回路への入力はファンクションジェネレータを使用し、入力周波数 1kHz で dc/SFQ セルに入力した。

A.2 測定結果

図 76 にバイアス電流に対する出力電圧特性を、図 77 にバイアス電流に対する出力電圧振幅特性を示す。図 76 において特性が傾いているのは、終端抵抗の 50Ω を介して観測しているためである。

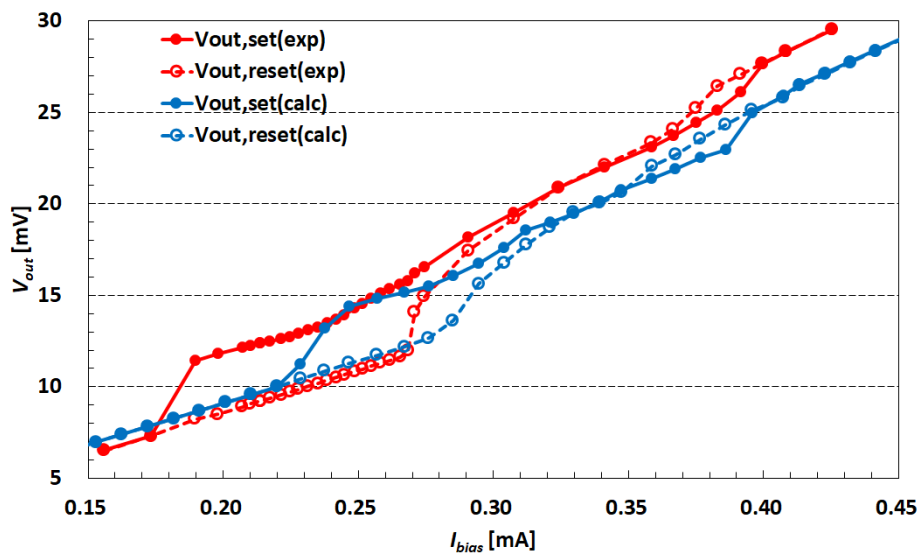


図 76 12 段単一 SQUID 構造分布型アンプのバイアス電流に対する出力電圧特性

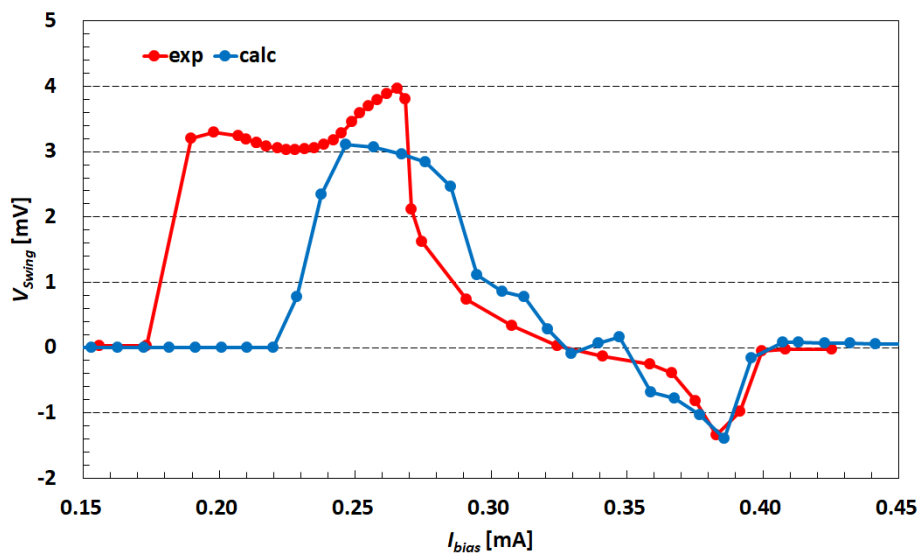


図 77 12 段単一 SQUID 構造分布型アンプのバイアス電流に対する出力電圧振幅特性

測定より、12 段単一 SQUID 構造分布型アンプの最大出力振幅は 3.97mV となった。数値計算では最大出力振幅が 3.10mV であるので、0.87mV だけ数値計算よりも出力電圧振幅が増加した。また、2.5mV 以上の出力電圧振幅が得られる領域を動作領域と仮定した場合、バイアス電流に対する動作領域は測定結果で $84.2\mu\text{A}$ 、数値計算で $45.5\mu\text{A}$ となり、測定結果は数値計算結果の 1.85 倍となった。

A.3 考察

測定結果の動作領域が数値計算よりも広がった原因を探るため、臨界電流値、結合係数を変化させた時に特性がどのように変化したかを調べた。

A.3.1 臨界電流値

最初に SQUID の臨界電流値を変化させた時に出力電圧特性がどのように変化するかを調べた。図 78 に SQUID の臨界電流値を 100 ~ 80% に 5% 刻みで変化させた時の結果を示す。

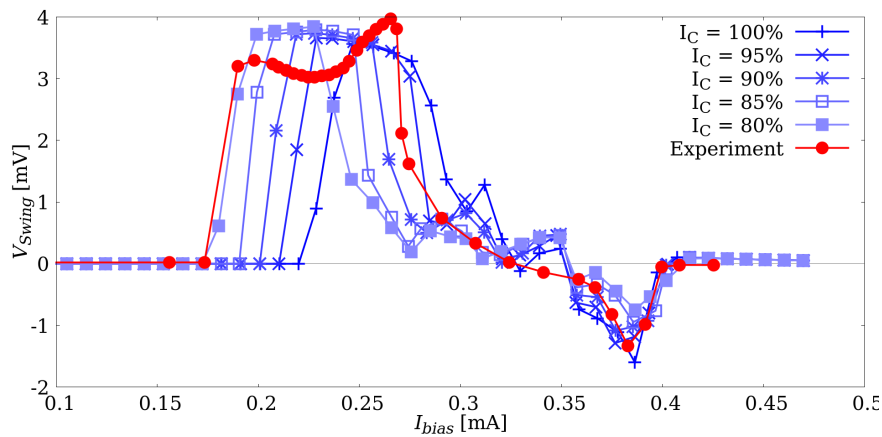


図 78 SQUID の臨界電流値を変化させた時の出力電圧特性の変化

図より、臨界電流値が減少するとその分動作領域全体が左にシフトすることが分かる。これは臨界電流値の減少によって SQUID 全体の臨界電流値も減少したためである。

A.3.2 結合係数

設計ではストレージループと SQUID との結合係数は 0.428 であり、その時の結果が図 76 と図 76 の結果である。この結合係数を変化 5% ずつ増加させた際の出力電圧振幅特性の変化を図 79 にバイアス電流に対する出力電圧振幅特性を示す。

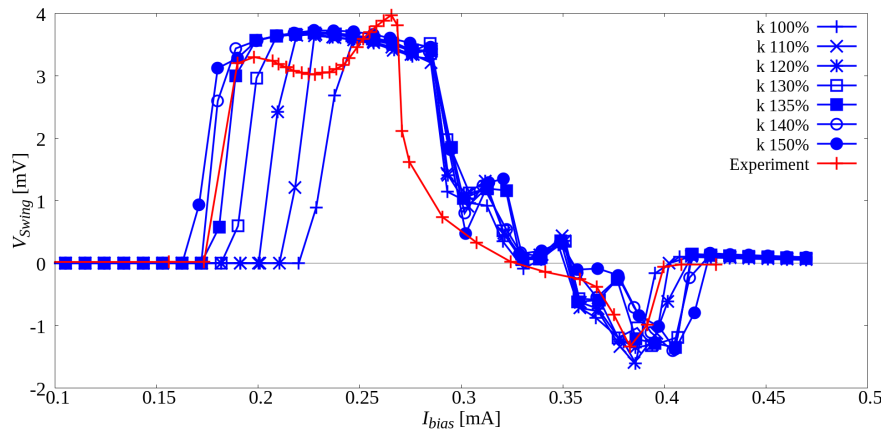


図 79 SQUID とストレージループの結合係数を変化させた時の出力電圧特性の変化

グラフの青線が結合係数を設計値を 100% としたときに 100 ~ 150% まで変化させた場合の結果であり、それぞれの点の記号が変化量に対応する。一方赤線は今回の測定で得られた特性である。図より、結合係数が設計値よりも 135% 高い値、つまり $k = 0.535$ の時に測定結果の特性と概ね一致していることが分かる。

結合係数が大きくなると、漏れ磁束の数が減少するためより少ない電流であっても SQUID が反応するようになるために結合係数の増加によって動作領域が増加したと考えられる。

付録 B 数値計算で用いた jsim ファイル

数値計算で示した JSIM ファイルを下記に示す。

4 段二重 SQUID 構造分布型アンプ (低速測定)

Listing 1 4 段二重 SQUID 構造分布型アンプ
の数値計算で使用した jsim ファイル

```

1 ****
2 *JSIM control file for CADENCE by
   kameda@cq.jp.nec.com
3 ****
4
5 *JSIM model
6 .model jjmod jj(Rtype=1, Vg=2.8mV, Cap
   =0.218pF, R0=200ohm, Rn=17ohm,
```

```

Icrit=0.1mA)
7
8 *** netlist file ***
9 ****
10 *** Lib : vswing_vs_freq
11 *** Cell:
   da_with_reset_4stage_dual_squid
12 *** View: schematic
13 *** Feb 21 12:56:27 2019
14 ****
15
16 *** ijtl
17 .subckt ijtl 1 2
```

```

18 *** din dout
19 L1 1 3 2.457pH fcheck
20 L3 3 2 1.422pH fcheck
21 LP1 4 0 0.073pH fcheck
22 B1 3 4 jjmod area=2.16
23 RS1 3 4 1.73ohm *SHUNT=3.73
24 .ends
25
26 *** jtl
27 .subckt jtl 1 2 5
28 *** din dout
29 R1 5 6 8.32ohm
30 LPIN 1 7 0.827pH fcheck
31 LPR1 6 7 4.451pH fcheck
32 L2 8 9 4.828pH fcheck
33 L3 9 2 1.422pH fcheck
34 L1 7 8 2.457pH fcheck
35 LP2 4 0 0.070pH fcheck
36 LP1 10 0 0.075pH fcheck
37 B2 9 4 jjmod area=2.16
38 RS2 9 4 1.73ohm *SHUNT=3.73
39 B1 8 10 jjmod area=2.16
40 RS1 8 10 1.73ohm *SHUNT=3.73
41 .ends
42
43 *** rtffb
44 .subckt rtffb 1 11 12 13 15
45 *** din dout0 dout1 reset
46 R1 15 16 8.34ohm
47 R3 15 17 12.18ohm
48 R2 15 18 8.34ohm
49 L9 19 11 1.323pH fcheck
50 L5 19 20 5.125pH fcheck
51 LP13 21 0 0.096pH fcheck
52 LX2 22 23 0.967pH fcheck
53 LP12 24 0 0.395pH fcheck
54 LP9 25 0 1.628pH fcheck
55 LP11 26 0 0.161pH fcheck
56 LP10 0 27 0.083pH fcheck
57 LP4 0 28 0.239pH fcheck
58 LP3 0 29 0.174pH fcheck
59 L10 20 22 4.521pH fcheck
60 LPIN1 13 30 0.772pH fcheck
61 LPR1 30 16 3.580pH fcheck
62 L8 30 31 2.540pH fcheck
63 L4 32 31 3.575pH fcheck
64 L7 33 12 1.495pH fcheck
65 L6 34 33 5.556pH fcheck

```

```

66 LX3 35 34 1.045pH fcheck
67 LEX4 36 37 1.443pH fcheck
68 L3 38 36 5.585pH fcheck
69 LEX3 39 36 1.414pH fcheck
70 LP1 40 0 0.278pH fcheck
71 LX1 23 38 1.061pH fcheck
72 LPR3 17 22 0.029pH fcheck
73 LEX1 41 42 1.672pH fcheck
74 L2 41 43 3.021pH fcheck
75 LEX2 44 41 1.310pH fcheck
76 LPR2 45 18 3.731pH fcheck
77 LPIN2 1 45 0.829pH fcheck
78 L1 45 44 2.168pH fcheck
79 B13 19 21 jjmod area=2.01
80 RS13 19 21 1.86ohm *SHUNT=3.73
81 B12 20 24 jjmod area=1.00
82 RS12 20 24 3.73ohm *SHUNT=3.73
83 B11 31 26 jjmod area=2.16
84 RS11 31 26 1.73ohm *SHUNT=3.73
85 B8 32 46 jjmod area=1.32
86 RS8 32 46 2.83ohm *SHUNT=3.73
87 B9 46 25 jjmod area=1.20
88 RS9 46 25 3.11ohm *SHUNT=3.73
89 B10 33 27 jjmod area=2.01
90 RS10 33 27 1.86ohm *SHUNT=3.73
91 B3 44 29 jjmod area=2.16
92 RS3 44 29 1.73ohm *SHUNT=3.73
93 B7 37 46 jjmod area=1.10
94 RS7 37 46 3.39ohm *SHUNT=3.73
95 B5 43 35 jjmod area=1.05
96 RS5 43 35 3.56ohm *SHUNT=3.73
97 B1 38 40 jjmod area=1.00
98 RS1 38 40 3.73ohm *SHUNT=3.73
99 B6 39 35 jjmod area=1.32
100 RS6 39 35 2.83ohm *SHUNT=3.73
101 B2 42 23 jjmod area=1.38
102 RS2 42 23 2.71ohm *SHUNT=3.73
103 B4 34 28 jjmod area=1.00
104 RS4 34 28 3.73ohm *SHUNT=3.73
105 .ends
106
107 *** jtlc
108 .subckt jtlc 1 2 5
109 *** din dout
110 R1 5 6 8.32ohm
111 LPIN 1 7 0.824pH fcheck
112 LPR1 6 7 4.235pH fcheck
113 L2 8 9 4.982pH fcheck

```

```

114 L3 9 2 1.555pH fcheck
115 L1 7 8 2.527pH fcheck
116 LP2 4 0 0.088pH fcheck
117 LP1 10 0 0.146pH fcheck
118 B2 9 4 jjmod area=2.16
119 RS2 9 4 1.73ohm *SHUNT=3.73
120 B1 8 10 jjmod area=2.16
121 RS1 8 10 1.73ohm *SHUNT=3.73
122 .ends
123
124 *** source
125 .subckt source 2
126 *** dout
127 Rd 16 0 4.02ohm
128 L2 10 16 4.917pH fcheck
129 L1 2 10 1.425pH fcheck
130 LP1 9 0 0.068pH fcheck
131 B1 10 9 jjmod area=2.16
132 RS1 10 9 1.73ohm *SHUNT=3.73
133 .ends
134
135 *** jtl3c
136 .subckt jtl3c 1 2 5
137 *** din dout
138 R2 5 48 16.65ohm
139 R1 5 6 8.34ohm
140 LPR2 48 9 1.209pH fcheck
141 L3 49 9 0.265pH fcheck
142 LP3 50 0 0.174pH fcheck
143 L5 51 2 1.518pH fcheck
144 LPIN 1 7 0.692pH fcheck
145 LPR1 6 7 5.983pH fcheck
146 L2 8 49 5.606pH fcheck
147 L4 9 51 4.742pH fcheck
148 L1 7 8 2.012pH fcheck
149 LP2 4 0 0.140pH fcheck
150 LP1 10 0 0.182pH fcheck
151 B3 51 50 jjmod area=2.16
152 RS3 51 50 1.73ohm *SHUNT=3.73
153 B2 49 4 jjmod area=2.16
154 RS2 49 4 1.73ohm *SHUNT=3.73
155 B1 8 10 jjmod area=2.16
156 RS1 8 10 1.73ohm *SHUNT=3.73
157 .ends
158
159 *** da_with_reset_dual_squid
160 .subckt da_with_reset_dual_squid 1 52 53
    2 48

```

```

161 *** din dinreset dinset dout
162 b9 54 55 jjmod area=2.03
163 *FIX
164 RS9 54 55 1.84ohm *SHUNT=3.73
165 b8 54 56 jjmod area=1.32
166 *FIX
167 RS8 54 56 2.83ohm *SHUNT=3.73
168 b7 45 57 jjmod area=1.32
169 *FIX
170 RS7 45 57 2.83ohm *SHUNT=3.73
171 b6 58 45 jjmod area=2.03
172 *FIX
173 RS6 58 45 1.84ohm *SHUNT=3.73
174 b4 59 60 jjmod area=1.00
175 *MAX= 1.443 *FIX
176 *RS4 59 60 4.24ohm *SHUNT=3.73
177 b5 61 62 jjmod area=1.00
178 *FIX
179 RS5 61 62 3.73ohm *SHUNT=3.73
180 b3 63 64 jjmod area=2.18
181 *FIX
182 RS3 63 64 1.71ohm *SHUNT=3.73
183 b2 65 6 jjmod area=2.18
184 *FIX
185 RS2 65 6 1.71ohm *SHUNT=3.73
186 b1 49 60 jjmod area=1.00
187 *FIX
188 RS1 49 60 3.73ohm *SHUNT=3.73
189 b0 66 62 jjmod area=1.00
190 *MAX= 1.443 *FIX
191 *RS0 66 62 4.24ohm *SHUNT=3.73
192 l29 55 65 2.000pH fcheck
193 l28 56 0 0.500pH fcheck *MIN= 0.050
194 l27 57 0 0.461pH fcheck *MIN= 0.050
195 l26 63 58 2.378pH fcheck
196 lsqc4 67 49 1.800pH fcheck *MIN= 0.320 *
    SLV=Lsqc3
197 Lsqc3 67 66 1.800pH fcheck *MIN= 0.320 *
    MST=Lsqc3
198 lbreset 68 69 3.500pH fcheck
199 lset2 70 63 2.000pH fcheck
200 lreset2 65 68 2.000pH fcheck
201 lset1 53 70 0.500pH fcheck
202 lbset 70 71 3.500pH fcheck
203 lreset1 68 52 0.500pH fcheck
204 lJ3 64 0 0.500pH fcheck *MIN= 0.050
205 lJ2 6 0 0.500pH fcheck *MIN= 0.050
206 lsig2 54 3 4.690pH fcheck *FIX *SLV=Lsig1

```



```

207 Lsig1 3 45 4.690pH fcheck *FIX *MST=
    Lsig1
208 lout 72 2 1.600pH fcheck
209 lin 1 67 2.400pH fcheck
210 Lsqc2 72 59 1.200pH fcheck *MIN= 0.320 *
    SLV=Lsqc1
211 Lsqc1 61 72 1.200pH fcheck *MIN= 0.320 *
    MST=Lsqc1
212 lsqb2 5 60 4.120pH fcheck *MIN= 0.050 *
    FIX *SLV=lsqb1
213 lsqb1 62 5 4.120pH fcheck *MIN= 0.050 *
    FIX *MST=lsqb1
214 K1 Lsig1 Lsqb1 0.730
215 K0 Lsig2 Lsqb2 0.730
216 rbreset 69 48 20.37ohm
217 rbset 71 48 17.27ohm
218 .ends
219
220 *** indct_125
221 .subckt indct_125 1 2
222 *** din dout
223 L0 1 2 125.000pH fcheck *FIX
224 C0 1 0 0.100pF *FIX
225 .ends
226
227 *** indct_250
228 .subckt indct_250 1 2
229 *** din dout
230 L0 1 2 250.000pH fcheck *FIX
231 C0 1 0 0.100pF *FIX
232 .ends
233
234 *** spll
235 .subckt spll 74 75 76 84
236 *** a b c
237 B1 77 78 jjmod area=2.15
238 RS1 77 78 1.74ohm *SHUNT=3.73
239 B3 79 80 jjmod area=1.94
240 RS3 79 80 1.92ohm *SHUNT=3.73
241 B2 25 81 jjmod area=1.94
242 RS2 25 81 1.92ohm *SHUNT=3.73
243 L4 25 75 1.191pH fcheck
244 LP1 78 0 0.153pH fcheck
245 LP3 80 0 0.122pH fcheck
246 L1 82 77 2.301pH fcheck
247 L5 79 76 1.191pH fcheck
248 L6 77 14 0.156pH fcheck
249 LPR1 83 82 4.173pH fcheck

250 LPA 74 82 0.822pH fcheck
251 L3 14 79 5.096pH fcheck
252 L2 14 25 5.218pH fcheck
253 LP2 81 0 0.239pH fcheck
254 R1 84 83 8.35ohm *FIX
255 .ends
256
257 *** cbg
258 .subckt cbg 74 75 76 18
259 *** a b c
260 B4 46 85 jjmod area=1.82
261 RS4 46 85 2.05ohm *SHUNT=3.73
262 B2 86 87 jjmod area=2.16
263 RS2 86 87 1.73ohm *SHUNT=3.73
264 B1 88 89 jjmod area=2.16
265 RS1 88 89 1.73ohm *SHUNT=3.73
266 B6 90 84 jjmod area=2.16
267 RS6 90 84 1.73ohm *SHUNT=3.73
268 B7 31 81 jjmod area=2.16
269 RS7 31 81 1.73ohm *SHUNT=3.73
270 B3 30 85 jjmod area=1.82
271 RS3 30 85 2.05ohm *SHUNT=3.73
272 B5 91 92 jjmod area=2.70
273 RS5 91 92 1.38ohm *SHUNT=3.73
274 L2 31 86 3.869pH fcheck
275 L6 86 46 1.162pH fcheck
276 LPR2 93 33 5.985pH fcheck
277 L9 94 31 1.841pH fcheck
278 L7 85 95 0.013pH fcheck
279 LP2 87 0 0.887pH fcheck
280 L4 91 76 0.655pH fcheck
281 LPR3 96 94 5.793pH fcheck
282 LP7 81 0 0.585pH fcheck
283 L1 90 88 3.817pH fcheck
284 LPIN2 75 94 0.829pH fcheck
285 L5 88 30 1.118pH fcheck
286 LP6 84 0 0.523pH fcheck
287 L8 33 90 1.859pH fcheck
288 L3 95 91 4.867pH fcheck
289 LP1 89 0 1.430pH fcheck
290 LPIN1 74 33 0.946pH fcheck
291 LPR1 97 95 0.177pH fcheck
292 LP5 92 0 0.322pH fcheck
293 R1 18 97 9.00ohm
294 R3 18 96 8.34ohm *FIX
295 R2 18 93 8.34ohm *FIX
296 .ends
297

```

```

298 *** top cell:
      da_with_reset_4stage_dual_squid
299 XI303 jtl 100 101
300 XI300 jtl 102 103
301 X9 jtl 104 105
302 XI352 jtl 106 107 171
303 XI349 jtl 108 40 171
304 XI347 jtl 109 110 171
305 XI346 jtl 111 112 171
306 XI345 jtl 113 114 171
307 XI343 jtl 115 116 171
308 XI340 jtl 36 117 171
309 XI338 jtl 118 119 171
310 XI336 jtl 120 47 171
311 XI208 jtl 121 122 171
312 XI333 jtl 123 120 171
313 XI311 jtl 124 125 171
314 XI310 jtl 125 126 171
315 XI309 jtl 127 128 171
316 XI308 jtl 129 127 171
317 XI307 jtl 130 129 171
318 XI306 jtl 131 130 171
319 XI305 jtl 132 131 171
320 XI304 jtl 101 132 171
321 XI299 jtl 103 133 171
322 XI298 jtl 133 134 171
323 XI297 jtl 134 135 171
324 XI296 jtl 135 136 171
325 XI295 jtl 136 137 171
326 XI294 jtl 137 138 171
327 XI292 jtl 139 140 171
328 XI291 jtl 140 141 171
329 XI280 jtl 62 142 171
330 XI204 jtl 143 144 171
331 XI325 jtl 145 146 171
332 XI176 jtl 105 147 171
333 XI126 jtl 21 22 171
334 XI125 jtl 22 148 171
335 XI123 jtl 47 149 171
336 XI116 jtl 51 150 171
337 XI115 jtl 151 152 171
338 XI238 jtl 50 153 171
339 XI402 jtl 147 16 171
340 XI415 jtl 16 154 171
341 XI425 jtl 155 78 171
342 XI180 jtl 156 17 171
343 XI435 jtl 157 158 171
344 XI316 jtl 67 159 171

```

```

345 XI441 jtl 158 21 171
346 XI290 jtl 141 160 171
347 XI458 jtl 77 4 171
348 XI459 jtl 161 8 171
349 *XI467 jtl 4 14 171
350 XI467 jtl 4 50 171
351 *XI468 jtl 8 162 171
352 XI468 jtl 8 121 171
353 XI433 rtffb 154 155 62 80 171
354 XI312 jtlc 128 124 171
355 XI293 jtlc 138 139 171
356 XI328 jtlc 150 145 171
357 XI327 jtlc 152 156 171
358 XI426 jtlc 78 157 171
359 XI547 jtlc 163 80 171
360 *R54 19 51 10.000fohm
361 *R53 164 151 10.000fohm
362 *R52 165 109 10.000fohm
363 *R51 166 111 10.000fohm
364 *R49 72 118 10.000fohm
365 *R39 167 113 10.000fohm
366 *R56 14 50 100.000pohm
367 *R55 162 121 100.000pohm
368 R26 168 169 1000.00ohm
369 R27 0 170 50.00ohm
370 V2 100 0 PULSE(0.0mV 0.035mV 2000.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
371 V1 102 0 PULSE(0.0mV 0.035mV 1000.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
372 VIN 104 0 PULSE(0.0mV 1.035mV 500.0ps
      1.0ps 1.0ps 1.0ps 1000.0ps)
373 XI448 source 163
374 XI434 jtl3c 142 123 171
375 Xda2 da_with_reset_dual_squid 172 107
      40 173 171
376 Xda1 da_with_reset_dual_squid 174 117
      116 175 171
377 Xda0 da_with_reset_dual_squid 176 159
      144 49 171
378 Xda3 da_with_reset_dual_squid 177 146
      17 178 171
379 XI337 indct_125 170 176
380 XI330 indct_125 178 169
381 vB 171 0 PWL(0ps 0mv 20ps 2.5mv) *FIX
382 vbsq 168 0 PWL(0ps 0mv 20ps 150mv) *
      FIX
383 XI351 indct_250 173 177
384 XI341 indct_250 175 172

```

```

385 XI120 indct_250 49 174
386 *XI353 spll 112 106 19 171
387 XI353 spll 112 106 51 171
388 *XI348 spll 110 108 164 171
389 XI348 spll 110 108 151 171
390 *XI344 spll 114 115 165 171
391 XI344 spll 114 115 109 171
392 *XI339 spll 119 36 166 171
393 XI339 spll 119 36 111 171
394 *XI239 spll 153 67 72 171
395 XI239 spll 153 67 118 171
396 *XI206 spll 122 143 167 171
397 XI206 spll 122 143 113 171
398 XI302 cbg 148 160 161 171
399 XI314 cbg 149 126 77 171
400
401 *** netlist file ***
402
403 *** jsim input file ***
404 * for scope
405 *.tran 1.5ps 2.2ns 0ps 0.2ps
406 * for operate check
407 .tran 0.5ps 5.0ns 0ps 0.2ps
408 .file OUT1
409 .print phase XDA0_B0 *2
410 .print phase XDA0_B1 *3

```

```

411 .print phase XDA0_B4 *4
412 .print phase XDA0_B5 *5
413 .print devi r26 *6
414 .print nodev 169 170 *7
415
416 .print phase XDA0_B0 *8
417 .print phase XDA0_B5 *9
418 .print phase XDA0_B4 *10
419 .print phase XDA0_B1 *11
420
421 .print phase XDA1_B0 *12
422 .print phase XDA1_B5 *13
423 .print phase XDA1_B4 *14
424 .print phase XDA1_B1 *15
425
426 .print phase XDA2_B0 *16
427 .print phase XDA2_B5 *17
428 .print phase XDA2_B4 *18
429 .print phase XDA2_B1 *19
430
431 .print phase XDA3_B0 *20
432 .print phase XDA3_B5 *21
433 .print phase XDA3_B4 *22
434 .print phase XDA3_B1 *23
435 *** jsim input file ***

```

12 段二重 SQUID 構造分布型アンプ (高速測定)

このファイルは高速測定用にパラメータを変更した後の JSIM ファイルになる。低速測定用に設計した 12 段回路は上のパラメータと同じなので省略する。

Listing 2 12 段二重 SQUID 構造分布型アンプ
の数値計算で使した jsim ファイル

```

1 *****
2 *JSIM control file for CADENCE by
   kameda@cq.jp.nec.com
3 *****
4
5 *JSIM model
6 .model jjmod jj(Rtype=1, Vg=2.8mV, Cap
   =0.218pF, R0=200ohm, Rn=17ohm,

```

```

Icrit=0.1mA)
7
8 *** netlist file ***
9 *****
10 *** Lib : OPEN572_khiguchi
11 *** Cell: DSDA_Reset_12stage_test
12 *** View: schematic
13 *** Nov 16 14:03:16 2018
14 *****
15
16 *** ijtl
17 .subckt ijtl 1 2
18 *** din dout

```

```

19 L1 1 3 2.457pH fcheck
20 L3 3 2 1.422pH fcheck
21 LP1 4 0 0.073pH fcheck
22 B1 3 4 jjmod area=2.16
23 RS1 3 4 1.73ohm *SHUNT=3.73
24 .ends
25
26 *** jtl
27 .subckt jtl 1 2 5
28 *** din dout
29 R1 5 6 8.32ohm
30 LPIN 1 7 0.827pH fcheck
31 LPR1 6 7 4.451pH fcheck
32 L2 8 9 4.828pH fcheck
33 L3 9 2 1.422pH fcheck
34 L1 7 8 2.457pH fcheck
35 LP2 4 0 0.070pH fcheck
36 LP1 10 0 0.075pH fcheck
37 B2 9 4 jjmod area=2.16
38 RS2 9 4 1.73ohm *SHUNT=3.73
39 B1 8 10 jjmod area=2.16
40 RS1 8 10 1.73ohm *SHUNT=3.73
41 .ends
42
43 *** indct_125
44 .subckt indct_125 1 2
45 *** din dout
46 L0 1 2 125.000pH fcheck *FIX
47 C0 1 0 0.100pF *FIX
48 .ends
49
50 *** DSDA_Matching
51 .subckt DSDA_Matching 1 11 12 2 35
52 *** din dinreset dinset dout
53 b9 13 14 jjmod area=2.03
54 *FIX
55 RS9 13 14 1.84ohm *SHUNT=3.73
56 b8 13 15 jjmod area=1.32
57 *FIX
58 RS8 13 15 2.83ohm *SHUNT=3.73
59 b7 16 17 jjmod area=1.32
60 *FIX
61 RS7 16 17 2.83ohm *SHUNT=3.73
62 b6 18 16 jjmod area=2.03
63 *FIX
64 RS6 18 16 1.84ohm *SHUNT=3.73
65 b4 19 20 jjmod area=1.00
66 *MAX= 1.443 *FIX

```

```

67 *RS4 19 20 4.24ohm *SHUNT=3.73
68 b5 21 22 jjmod area=1.00
69 *FIX
70 RS5 21 22 3.73ohm *SHUNT=3.73
71 b3 23 24 jjmod area=2.18
72 *FIX
73 RS3 23 24 1.71ohm *SHUNT=3.73
74 b2 25 6 jjmod area=2.18
75 *FIX
76 RS2 25 6 1.71ohm *SHUNT=3.73
77 b1 26 20 jjmod area=1.00
78 *FIX
79 RS1 26 20 3.73ohm *SHUNT=3.73
80 b0 27 22 jjmod area=1.00
81 *MAX= 1.443 *FIX
82 *RS0 27 22 4.24ohm *SHUNT=3.73
83 l29 14 25 2.000pH fcheck
84 l28 15 0 0.500pH fcheck *MIN= 0.050
85 l27 17 0 0.461pH fcheck *MIN= 0.050
86 l26 23 18 2.378pH fcheck
87 *lsqc4 28 26 1.800pH fcheck *MIN= 0.320 *
    SLV=Lsqc3
88 *Lsqc3 28 27 1.800pH fcheck *MIN= 0.320
    *MST=Lsqc3
89 lsqc4 28 26 0.500pH fcheck *MIN= 0.320 *
    SLV=Lsqc3
90 Lsqc3 28 27 0.500pH fcheck *MIN= 0.320 *
    MST=Lsqc3
91 lbreset 29 30 3.500pH fcheck
92 lset2 31 23 2.000pH fcheck
93 lreset2 25 29 2.000pH fcheck
94 lset1 12 31 0.500pH fcheck
95 lbset 31 32 3.500pH fcheck
96 lreset1 29 11 0.500pH fcheck
97 lJ3 24 0 0.500pH fcheck *MIN= 0.050
98 lJ2 6 0 0.500pH fcheck *MIN= 0.050
99 *lsig2 13 3 4.690pH fcheck *fix *slv=lsig1
100 *lsig1 3 16 4.690pH fcheck *fix *mst=lsig1
101 lsig2 13 3 6.690pH fcheck *fix *slv=lsig1
102 lsig1 3 16 6.690pH fcheck *fix *mst=lsig1
103 lout 33 2 1.600pH fcheck
104 lin 1 28 2.400pH fcheck
105 *Lsqc2 33 19 1.200pH fcheck *MIN= 0.320
    *SLV=Lsqc1
106 *Lsqc1 21 33 1.200pH fcheck *MIN= 0.320
    *MST=Lsqc1
107 Lsqc2 33 19 0.500pH fcheck *MIN= 0.320 *
    SLV=Lsqc1

```

```

108 Lsqc1 21 33 0.500pH fcheck *MIN= 0.320 *
    MST=Lsqc1
109 *lsqb2 5 20 4.120pH fcheck *MIN= 0.050 *
    FIX *SLV=Lsqb1
110 *Lsqb1 22 5 4.120pH fcheck *MIN= 0.050 *
    FIX *MST=Lsqb1
111 lsqb2 5 20 3.120pH fcheck *MIN= 0.050 *
    FIX *SLV=Lsqb1
112 Lsqb1 22 5 3.120pH fcheck *MIN= 0.050 *
    FIX *MST=Lsqb1
113 K1 Lsig1 Lsqb1 0.730
114 K0 Lsig2 Lsqb2 0.730
115 rbreset 30 35 20.37ohm
116 rbset 32 35 17.27ohm
117 .ends
118
119 *** indct_250
120 .subckt indct_250 1 2
121 *** din dout
122 L0 1 2 250.000pH fcheck *FIX
123 C0 1 0 0.100pF *FIX
124 .ends
125
126 *** top cell: DSDA_Reset_12stage_test
127 XI543 ijtl 36 37
128 XI542 ijtl 38 39
129 XI541 ijtl 40 41
130 XI540 ijtl 42 43
131 XI539 ijtl 33 44
132 XI538 ijtl 26 28
133 XI537 ijtl 45 46
134 XI536 ijtl 47 48
135 XI549 ijtl 35 49
136 XI548 ijtl 50 51
137 XI535 ijtl 24 52
138 XI534 ijtl 53 54
139 XI533 ijtl 55 56
140 XI547 ijtl 57 58
141 XI544 ijtl 59 60
142 XI546 ijtl 61 62
143 XI532 ijtl 63 64
144 XI531 ijtl 65 66
145 XI530 ijtl 67 68
146 XI529 ijtl 69 70
147 XI545 ijtl 71 72
148 XI528 ijtl 73 74
149 XI527 ijtl 75 76
150 XI526 ijtl 77 78
151 XI484 jtl 79 80 150
152 XI483 jtl 81 82 150
153 XI524 jtl 83 25 150
154 XI523 jtl 64 83 150
155 XI508 jtl 84 85 150
156 XI507 jtl 86 87 150
157 XI496 jtl 88 89 150
158 XI495 jtl 90 91 150
159 XI482 jtl 60 81 150
160 XI481 jtl 72 92 150
161 XI480 jtl 92 93 150
162 XI479 jtl 93 94 150
163 XI494 jtl 62 90 150
164 XI493 jtl 58 95 150
165 XI506 jtl 46 86 150
166 XI505 jtl 68 96 150
167 XI522 jtl 56 97 150
168 XI521 jtl 97 98 150
169 XI520 jtl 98 99 150
170 XI519 jtl 100 101 150
171 XI504 jtl 48 102 150
172 XI503 jtl 102 103 150
173 XI492 jtl 95 104 150
174 XI491 jtl 89 105 150
175 XI478 jtl 106 107 150
176 XI477 jtl 108 109 150
177 XI476 jtl 110 108 150
178 XI475 jtl 74 110 150
179 XI490 jtl 91 111 150
180 XI489 jtl 80 112 150
181 XI502 jtl 103 113 150
182 XI501 jtl 104 114 150
183 XI518 jtl 52 100 150
184 XI517 jtl 101 115 150
185 XI516 jtl 54 116 150
186 XI515 jtl 116 117 150
187 XI500 jtl 118 119 150
188 XI499 jtl 120 118 150
189 XI488 jtl 82 121 150
190 XI487 jtl 122 106 150
191 XI474 jtl 123 124 150
192 XI509 jtl 66 84 150
193 XI473 jtl 125 123 150
194 XI498 jtl 49 120 150
195 XI472 jtl 41 125 150
196 XI486 jtl 44 122 150
197 XI460 jtl 13 126 150
198 XI459 jtl 127 128 150

```

```

199 XI464 jtl 129 130 150
200 XI465 jtl 43 129 150
201 XI466 jtl 23 131 150
202 XI463 jtl 132 133 150
203 XI458 jtl 130 134 150
204 XI457 jtl 135 136 150
205 XI456 jtl 133 137 150
206 XI455 jtl 131 138 150
207 XI454 jtl 126 139 150
208 XI462 jtl 39 132 150
209 XI461 jtl 37 13 150
210 XI510 jtl 96 140 150
211 XI497 jtl 51 88 150
212 XI471 jtl 78 23 150
213 XI470 jtl 70 141 150
214 XI485 jtl 28 79 150
215 XI514 jtl 117 142 150
216 XI511 jtl 87 143 150
217 XI512 jtl 85 144 150
218 XI513 jtl 140 145 150
219 XI467 jtl 76 16 150
220 XI469 jtl 141 127 150
221 XI468 jtl 16 135 150
222 XI525 jtl 25 146 150
223 R51 147 0 50.00ohm
224 R26 148 147 1000.00ohm
225 R27 0 149 50.00ohm
226
227 *VIN 105 0 PULSE(0.0mV 1.035mV 1000.0
      ps 1.0ps 1.0ps 1.0ps 1000.0ps)
228 *** set
229 V1 36 0 PULSE(0.0mV 1.035mV 1000.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
230 V14 38 0 PULSE(0.0mV 1.035mV 1005.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
231 V15 42 0 PULSE(0.0mV 1.035mV 1010.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
232 V16 40 0 PULSE(0.0mV 1.035mV 1015.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
233 V17 33 0 PULSE(0.0mV 1.035mV 1020.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
234 V18 26 0 PULSE(0.0mV 1.035mV 1025.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
235 V19 59 0 PULSE(0.0mV 1.035mV 1030.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
236 V20 71 0 PULSE(0.0mV 1.035mV 1035.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
237 V21 57 0 PULSE(0.0mV 1.035mV 1040.0ps

```

```

      1.0ps 1.0ps 1.0ps 2000.0ps)
238 V22 61 0 PULSE(0.0mV 1.035mV 1045.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
239 V23 50 0 PULSE(0.0mV 1.035mV 1050.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
240 V24 35 0 PULSE(0.0mV 1.035mV 1055.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
241 *** reset
242 V2 77 0 PULSE(0.0mV 1.035mV 2000.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
243 V3 75 0 PULSE(0.0mV 1.035mV 2005.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
244 V5 69 0 PULSE(0.0mV 1.035mV 2010.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
245 V4 73 0 PULSE(0.0mV 1.035mV 2015.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
246 V9 67 0 PULSE(0.0mV 1.035mV 2020.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
247 V8 65 0 PULSE(0.0mV 1.035mV 2025.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
248 V6 45 0 PULSE(0.0mV 1.035mV 2030.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
249 V7 47 0 PULSE(0.0mV 1.035mV 2035.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
250 V10 53 0 PULSE(0.0mV 1.035mV 2040.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
251 V11 24 0 PULSE(0.0mV 1.035mV 2045.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
252 V12 55 0 PULSE(0.0mV 1.035mV 2050.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
253 V13 63 0 PULSE(0.0mV 1.035mV 2055.0ps
      1.0ps 1.0ps 1.0ps 2000.0ps)
254 ***
255 vB 150 0 PWL(0ps 0mv 20ps 2.5mv) *FIX
256 vbsq 148 0 PWL(0ps 0mv 20ps 200mv) *
      FIX
257 XI413 indct_125 151 147
258 XI329 indct_125 149 152
259 Xda9 DSDA_Matching 153 115 111 154 150
260 Xda8 DSDA_Matching 155 142 114 156 150
261 Xda10 DSDA_Matching 157 99 105 158 150
262 Xda11 DSDA_Matching 159 146 119 151
      150
263 Xda5 DSDA_Matching 160 144 112 161 150
264 Xda4 DSDA_Matching 162 145 107 163 150
265 Xda6 DSDA_Matching 164 143 121 31 150
266 Xda7 DSDA_Matching 165 113 94 166 150
267 Xda3 DSDA_Matching 167 109 124 168 150

```

```

268 Xda2 DSDA_Matching 169 128 134 5 150
269 Xda1 DSDA_Matching 170 136 137 29 150
270 Xda0 DSDA_Matching 152 138 139 171 150
271 XI451 indct_250 161 164
272 XI450 indct_250 166 155
273 XI430 indct_250 156 153
274 XI414 indct_250 158 159
275 XI449 indct_250 154 157
276 XI453 indct_250 29 169
277 XI392 indct_250 163 160
278 XI379 indct_250 31 165
279 XI452 indct_250 168 162
280 XI358 indct_250 5 167
281 XI338 indct_250 171 170
282
283 *** netlist file ***
284
285 *** jsim input file ***
286 * for scope
287 *.tran 1.5ps 2.2ns 0ps 0.2ps
288 * for operate check
289 .tran 0.5ps 3.5ns 0ps 0.2ps
290 .file trial_25th.out
291 .print phase XDA0_B0 *2
292 .print phase XDA0_B1 *3
293 .print phase XDA0_B4 *4
294 .print phase XDA0_B5 *5
295 .print devi r26 *6
296 .print nodev 147 149 *7
297
298 .print phase XDA0_B0 *8
299 .print phase XDA0_B1 *9
300 .print phase XDA0_B4 *10
301 .print phase XDA0_B5 *11
302
303 .print phase XDA1_B0 *
304 .print phase XDA1_B1 *
305 .print phase XDA1_B4 *
306 .print phase XDA1_B5 *15
307
308 .print phase XDA2_B0 *
309 .print phase XDA2_B1 *
310 .print phase XDA2_B4 *
311 .print phase XDA2_B5 *
312
313 .print phase XDA3_B0 *20
314 .print phase XDA3_B1 *
315 .print phase XDA3_B4 *
316 .print phase XDA3_B5 *
317
318 .print phase XDA4_B0 *
319 .print phase XDA4_B1 *25
320 .print phase XDA4_B4 *
321 .print phase XDA4_B5 *
322
323 .print phase XDA5_B0 *
324 .print phase XDA5_B1 *
325 .print phase XDA5_B4 *30
326 .print phase XDA5_B5 *
327
328 .print phase XDA6_B0 *
329 .print phase XDA6_B1 *
330 .print phase XDA6_B4 *
331 .print phase XDA6_B5 *35
332
333 .print phase XDA7_B0 *
334 .print phase XDA7_B1 *
335 .print phase XDA7_B4 *
336 .print phase XDA7_B5 *
337
338 .print phase XDA8_B0 *40
339 .print phase XDA8_B1 *
340 .print phase XDA8_B4 *
341 .print phase XDA8_B5 *
342
343 .print phase XDA9_B0 *
344 .print phase XDA9_B1 *45
345 .print phase XDA9_B4 *
346 .print phase XDA9_B5 *
347
348 .print phase XDA10_B0 *
349 .print phase XDA10_B1 *
350 .print phase XDA10_B4 *50
351 .print phase XDA10_B5 *
352
353 .print phase XDA11_B0 *
354 .print phase XDA11_B1 *
355 .print phase XDA11_B4 *
356 .print phase XDA11_B5 *55
357
358 *** output voltage each stage ***
359 .print nodev 153 154 *
360 .print nodev 155 156 *
361 .print nodev 157 158 *
362 .print nodev 159 151 *
363 .print nodev 160 161 *60

```

```

364 .print nodev 162 163 *
365 .print nodev 164 31 *
366 .print nodev 165 166 *
367 .print nodev 167 168 *
368 .print nodev 169 5 *65
369 .print nodev 170 29 *
370 .print nodev 152 171 *
371
372 *** sfq delay set
373 .print phase XI454_B2 *
374 .print phase XI456_B2 *
375 .print phase XI458_B2 *70
376 .print phase XI474_B2 *
377 .print phase XI478_B2 *
378 .print phase XI489_B2 *
379 .print phase XI488_B2 *
380 .print phase XI479_B2 *75
381 .print phase XI501_B2 *
382 .print phase XI490_B2 *
383 .print phase XI491_B2 *
384 .print phase XI500_B2 *
385
386 *** sfq delay reset
387 .print phase XI455_B2 *80
388 .print phase XI457_B2 *
389 .print phase XI459_B2 *

390 .print phase XI477_B2 *
391 .print phase XI513_B2 *
392 .print phase XI512_B2 *85
393 .print phase XI511_B2 *
394 .print phase XI502_B2 *
395 .print phase XI514_B2 *
396 .print phase XI517_B2 *
397 .print phase XI520_B2 *90
398 .print phase XI525_B2 *
399
400 *** loop current in SQUID
401 .print devi Xda0_lsqb1 *
402 .print devi Xda1_lsqb1 *
403 .print devi Xda2_lsqb1 *
404 .print devi Xda3_lsqb1 *95
405 .print devi Xda6_lsqb1 *
406 .print devi Xda7_lsqb1 *
407 .print devi Xda5_lsqb1 *
408 .print devi Xda4_lsqb1 *
409 .print devi Xda10_lsqb1 *100
410 .print devi Xda11_lsqb1 *
411 .print devi Xda9_lsqb1 *
412 .print devi Xda8_lsqb1 *103
413
414 *** jsim input file ***

```

付録 C 数値計算で用いた補助プログラム

バイアス電流-出力電圧振幅を求めるために、バイアス電流を一定値ずつ変化させながら JSIM を実行するコードと、その結果から特性を求めるコードを示す。2つの機能があるが、1つのファイルにまとまっている。

Listing 3 バイアス電流に対する特性を計算するコード

```

1 #!/usr/bin/python -u
2 # -*- coding: utf-8 -*-
3 """ を用いた数値計算の補助プログラム jsim
4 """
5
6 from __future__ import print_function
7 import os
8 import argparse

```



```

9  import tempfile
10 import re
11 import readline
12 import subprocess
13 import csv
14 import shutil
15 import datetime
16 import math
17 import sys, traceback # 時の詳細表示用 Exception
18 from logging import getLogger, StreamHandler, DEBUG, Formatter, FileHandler
19
20 logger = getLogger('calc_margin')
21 logger.setLevel(DEBUG)
22 handler_format = Formatter('[%(levelname)s : %(asctime)s] : %(message)s')
23
24 stream_handler = StreamHandler()
25 stream_handler.setFormatter(handler_format)
26 stream_handler.setLevel(DEBUG)
27
28 file_handler = FileHandler('{ }.log'.format('calc_margin'), 'a')
29 file_handler.setFormatter(handler_format)
30 file_handler.setLevel(DEBUG)
31
32 logger.addHandler(stream_handler)
33 logger.addHandler(file_handler)
34 #logger.propagate = False
35
36 class CalcMargin:
37     """ で数値計算を行い、出力電圧を計算・出力するためのクラス jsim
38     """
39     def __init__(self):
40         """ クラスの初期化を行うメソッド
41         Notes:による計算を何回行うかや電圧の計算に必要な変数などを定義、初期化している
42             jsim
43
44         """
45         self.jsim_input = ""
46         self.bias_start = 0

```

```

47     self.bias_end = 0
48     self.now_current = 0
49     self.output_file = ""
50     self.jsim_temp = tempfile.mkstemp(prefix='jsim_temp.')
51     self.bias_step = 0
52     self.stages = 0
53     self.kj = 3.29109e-16 # Josephson statics
54     self.calc_voltage = [0, 0] # 0: b0, 1: b1
55     self.calc_voltage2 = [0, 0] # 0: b0, 1: b1
56     self.calc_current = 0 # current voltage of Ibias
57     self.calc_minmax = [0, 0] # 0: max, 1: min
58     # 遅延時間に関する記述
59     # TODO: 段で決め打ちしてしまっているの、他の段でも適用できるようにする 12
60     # 回路の構成によって遅延時間は異なってくるため、段数やの数と同じようにファイル
61     #   を用意して記述させる方式にした方が良さそう JJconfig
62     self.jtl_delay = 8.3e-12 # の伝搬遅延 JTL [s]
63     self.sp1l_delay = 10.1e-12 # の伝搬遅延 Splitter [s]
64     self.jj_delay = 3.7e-12 # の伝搬遅延 JJ [s]
65     self.rtffb0_delay = 17.2e-12 # rtffb(dout0)の伝搬遅延 [s]
66     self.rtffb1_delay = 16.0e-12 # rtffb(dout1)の伝搬遅延 [s]
67     self.pulse_set = 0 # パルスの信号が最初に入力される時間 Set [ps]
68     self.pulse_reset = 0 # パルスの信号が最初に入力される時間 Reset [ps]
69     self.pulse_interval = 0 # パルスの入力感覚[ps]
70     self.wait_time = 0.3e-9 # 電圧計算の処理を始めるまでに待つ時間[s]
71     # Set
72     self.tau_set_first = self.jtl_delay*15 + self.rtffb0_delay + self.sp1l_delay # 段目の
73     #   1に入力されるまでの伝搬遅延 SQUID回路によって変わるので適宜変更しないとい
74     #   けない ()
75     self.tau_set_last = self.jtl_delay*27 + self.rtffb0_delay + self.sp1l_delay*11 # 段目
76     #   の 12に入力されるまでの伝搬遅延 SQUID
77     # Reset
78     self.tau_reset_first = self.jtl_delay*16 + self.rtffb1_delay + self.sp1l_delay # 段目
79     #   の 1に入力されるまでの伝搬遅延 SQUID
80     self.tau_reset_last = self.jtl_delay*28 + self.rtffb1_delay + self.sp1l_delay*11 # 段
81     #   目の 12に入力されるまでの伝搬遅延 SQUID
82     self.tau_squid = self.jj_delay*2 # での接合の立ち上がり時間 SQUID(JJ*2)
83
84     def parseArgument(self):
85         """ コマンドライン引数の読み取り
86         """

```

```

81     parser = argparse.ArgumentParser(description = "Calculate I-V characteristics at
           SQUID")
82     parser.add_argument("--input", type=str, help="input jsim file. [required]", required
           =True)
83     parser.add_argument("--bias_start", type=str, help="inital value of bias current. [
           required]", required=True)
84     parser.add_argument("--bias_end", type=str, help="inital value of bias current. [
           required]", required=True)
85     parser.add_argument("--bias_step", type=str, help="step value of bias current. [
           required]", required=True)
86     parser.add_argument("--stages", type=str, help="stages of distributed amplifier [
           required]", required=True)
87
88     args = parser.parse_args()
89
90     self.jsim_input = args.input
91     if not os.path.exists(self.jsim_input):
92         raise NameError("Input file is not exists. Did you type wrong file name?")
93
94     self.bias_start = int(args.bias_start)
95     self.bias_step = int(args.bias_step)
96     self.bias_end = int(args.bias_end)
97     self.stages = int(args.stages)
98
99     logger.debug("{} , {} , {} , {}".format(self.jsim_input, self.bias_start, self.bias_step, (self.
           bias_start - self.bias_end) / 2 + 1))
100
101     # 元となるファイルからバイアス電流を変更した新しいファイルを作成する jsimjsim
102     def modify_file(self):
103         """ 大元の用ファイルからバイアス電流、ファイル名を変更した新しい用ファイルを作成
           する jsimjsim
104
105         Note:大元となるファイルは各自用意することそのままに投げて動く状態で
106
107             jsim変更する素子名を
108             "v1", "vbsq"などで直打ちしているので変わっていれば適宜修正する "
109         TODO:直打ちをやめてファイル内の
110             Magic で判別するようにする Character

```

```

111     """
112     # 出力側からのバイアス電流
113     pat_volt = r"^vbsq.*PWL(. * . * .*)$"
114     re_volt = re.compile(pat_volt, re.IGNORECASE)
115     # 入力周波数
116     pat_freq = r"^VIN.*PULSE(.*)"
117     re_freq = re.compile(pat_freq, re.IGNORECASE)
118     # ファイル名
119     pat_file = r"^\.file OUT1*$"
120     re_file = re.compile(pat_file, re.IGNORECASE)
121     # 抵抗
122     pat_ohm = r'^r.*$'
123     re_ohm = re.compile(pat_ohm, re.IGNORECASE)
124     # 臨界電流
125     pat_ic = r'^b.*$'
126     re_ic = re.compile(pat_ic, re.IGNORECASE)
127     # L
128     pat_l = r'^l.*$'
129     re_l = re.compile(pat_l, re.IGNORECASE)
130     # k
131     pat_k = r'^k.*$'
132     re_k = re.compile(pat_k, re.IGNORECASE)
133
134     logger.debug("Modifying bias current in jsim file.....")
135     src = open(self.jsim_input, "r")
136     dst = open(self.jsim_temp[1], "w")
137     line = 1 # ファイルの行数指定用
138     for i in src.readlines():
139         buf = i
140         # 出力側からのバイアス電流
141         matchOB = re_volt.match(i.strip())
142         matchOB = None
143         if matchOB != None:
144             logger.debug(matchOB.group())
145             lists = matchOB.group().split()
146             lists[-2] = "{}mv".format(self.now_current)
147             buf = " ".join(lists) + "\n"
148             logger.debug(buf)

```

```

149     # 入力周波数 (をで割る now_current100)
150     matchOB = re_freq.match(i.strip())
151     if matchOB != None:
152         logger.debug(matchOB.group())
153         lists = matchOB.group().split()
154         freq = 1/(self.now_current*1e-2*1e9)*1e12
155         logger.debug(self.now_current*1e-2)
156         lists[-1] = "{}ps".format(freq)
157         buf = " ".join(lists) + "\n"
158         logger.debug(buf)
159     # ファイル名
160     matchOB = re_file.match(i.strip())
161     if matchOB != None:
162         logger.debug (matchOB.group())
163         buf = ".file {} \n".format(self.output_file)
164         logger.debug(buf)
165     # 抵抗
166     matchOB = re_ohm.match(i.strip())
167     matchOB = None
168     if matchOB != None: # and (line >= 204 and line <= 235):
169         factor = 1.10
170         lists = matchOB.group().split()
171         digit, suffix = re.findall(r'([\d.]+\D+)', lists[3])
172         lists[3] = "{}{}".format(float(digit)*factor, suffix)
173         buf = " ".join(lists) + "\n"
174         logger.debug(buf)
175     # 臨界電流行指定しているのはアンプの部分だけ臨界電流を変化させたいので ()
176     matchOB = re_ic.match(i.strip())
177     matchOB = None
178     if matchOB != None: # and (line >= 204 and line <= 235):
179         factor = 0.75
180         lists = matchOB.group().split()
181         logger.debug(lists[4])
182         lists[4] = "area={}".format(float(re.sub(r'[a-zA-Z=]', '', lists[4]))*factor)
183         buf = " ".join(lists) + "\n"
184         logger.debug(buf)
185     # L
186     matchOB = re_l.match(i.strip())

```

```

187         matchOB = None
188         if matchOB != None and (line >= 120 and line <= 128):
189             factor = 1.000 #65
190             lists = matchOB.group().split()
191             lists[3] = "{}pH".format(float(re.sub(r'[a-zA-Z=]', '', lists[3]))*factor)
192             buf = " ".join(lists) + "\n"
193             logger.debug(buf)
194         # k
195         matchOB = re_k.match(i.strip())
196         matchOB = None
197         if matchOB != None and (line >= 83 and line <= 150):
198             factor = 1.000
199             lists = matchOB.group().split()
200             lists[3] = "{}".format(float(re.sub(r'[a-zA-Z=]', '', lists[3]))*factor)
201             buf = " ".join(lists) + "\n"
202             logger.debug(buf)
203
204         dst.write(buf)
205         line += 1
206
207     dst.close()
208     src.close()
209
210     def decide_prefix(self, prefix):
211         """ 接頭辞を判定し、それに応じた累乗値を返す
212
213         Args:
214             prefix (string): 接頭辞を含む文字列
215
216         Notes:対応する接頭辞は
217             "ps", "ns", "us", "ms"のみ "これ以外の接頭辞は使う予定が無いので対応予定無し
218
219
220         Returns:対応する接頭辞が含まれていればそれに応じた累乗値を、無ければを返す
221             1
222
223         """
224         if prefix == "ps" or prefix == "PS":

```

```

225         return 1e-12
226     elif prefix == "ns" or prefix == "NS":
227         return 1e-9
228     elif prefix == "us" or prefix == "US":
229         return 1e-6
230     elif prefix == "ms" or prefix == "MS":
231         return 1e-3
232     else:
233         return 1
234
235     def get_pulse_time(self):
236         """ パルスの入力タイミング、周期を抽出する
237         """
238         logger.debug("get_pulse_time")
239         src = open(self.jsim_temp[1], "r")
240         patvin = r"^*?VIN[0-9\s]*PULSE\((.*)\$"
241         repvin = re.compile(patvin, re.IGNORECASE)
242         for i in src.readlines():
243             rere = re.findall(repvin, i.strip())
244             if rere:
245                 relistvin = rere[0].split()
246
247                 prefix = self.decide_prefix(relistvin[2][re.search(r'[a-zA-Z]', relistvin[2]).start():])
248                 self.pulse_set = float(re.sub(r'[a-zA-Z]', '', relistvin[2]))*prefix
249                 self.pulse_interval = float(re.sub(r'[a-zA-Z]', '', relistvin[6]))*prefix
250                 self.pulse_reset = self.pulse_set + self.pulse_interval
251                 logger.debug("-----")
252                 logger.debug("Set Reset Interval prefix")
253                 logger.debug("{} {} {} {}".format(self.pulse_set, self.pulse_reset, self.pulse_interval,
254                                                     prefix))
254                 src.close()
255
256     def get_simulate_time(self):
257         """ シミュレーション時間を抽出する
258         """
259         logger.debug("get_simulate_time")
260         src = open(self.jsim_temp[1], "r")
261         patsim = r"^\.tran(.*)\$"

```

```

262     resim = re.compile(patsim, re.IGNORECASE)
263     for i in src.readlines():
264         res = re.findall(resim, i.strip())
265         if res:
266             trans = res[0].split()
267
268     prefix = self.decide_prefix(trans[1][re.search(r'[a-zA-Z]', trans[1]).start():])
269     self.sim_time = float(re.sub(r'[a-zA-Z]', '', trans[1]))*prefix
270     logger.debug("Simulation time: {}".format(self.sim_time))
271     src.close()
272
273     def calculate_with_noise(self, temperature=4.2):
274         """ ノイズを負荷して数値計算を行う
275
276         Args:
277             temperature (int): 環境温度雑音の大きさを決めるのに使用 ()
278
279         Notes: 雑音の大きさは、
280              $\sqrt{4 * k_B * T / R}$  で決まるはボルツマン定数、は環境温度、は抵抗
281              $k_B T R$  この関数では別に用意されているファイルでファイルの全ての抵抗に対して
282             並列に雑音電流源を付加している
283             awkjsim もし付加したくない抵抗がある場合は、その部分をに置き換える
284             Z正常に数値計算が出来なかった場合は例外が飛ぶ
285             NameError
286
287         Returns: なし
288
289         """
290         FNULL = open(os.devnull, 'w')
291         args = ['awk', '-f', 'noise.awkf', 'temperature={}'.format(temperature), '{}'.format(
292             self.jsim_temp[1])]
293         res1 = subprocess.Popen(args, stdout=subprocess.PIPE)
294         args = ['jsim_n']
295         res2 = subprocess.Popen(args, stdin=res1.stdout)
296         res1.stdout.close()
297         res2.wait()
298         FNULL.close()
299         if os.path.getsize(self.output_file.upper()) == 0 or not os.path.exists(self.output_file.

```



```

        upper()):
298         raise NameError("jsim output file is not found or empty. It might be error
            occured in jsim.")

299
300     def calculate(self):
301         """ 数値計算を行う
302
303         Notes:正常に数値計算が出来なかった場合は例外が飛ぶ
304             NameError
305
306         """
307         FNULL = open(os.devnull, 'w')
308         args = ['jsim', '{}'.format(self.jsim_temp[1])]
309         res = subprocess.call(args, stdout=FNULL)
310         FNULL.close()
311         if os.path.getsize(self.output_file.upper()) == 0 or not os.path.exists(self.output_file.
            upper()):
312             raise NameError("jsim output file is not found or empty. It might be error
                occured in jsim.")
313
314     def makeIV(self):
315         """ 数値計算の結果からのを計算する SQUIDIV
316
317         Notes:位相の時間微分を求め、ジョセフソン定数を掛けることで求めている
318
319         """
320         with open(self.output_file.upper(), "r") as f:
321             out1 = f.readlines()
322             head = out1[0].strip().split(" ")
323             tail = out1[-1].strip().split(" ")
324             #logger.debug(["time", "ibias", "phase_b0", " phase_b1"])
325             #logger.debug(head)
326             #logger.debug(tail)
327             # voltage at B0
328             #self.calc_voltage[0] = self.kj * (float(tail[1]) - float(head[1])) / (float(tail[0]) - float(
                head[0]))
329             # voltage at B1
330             #self.calc_voltage[1] = self.kj * (float(tail[2]) - float(head[2])) / (float(tail[0]) - float(

```

```

        head[0]))
331     # バイアス電流値本当は自動で判別したい ()
332     self.calc_current = float(tail[5])
333
334     def makeVoltage_renew(self):
335         """ 数値計算の結果から、状態の電圧を計算する SETRESET
336
337         Returns:形式で
338             tuple(状態の電圧 SET, 状態の電圧 RESET, 振幅の最大値, 振幅の最小値)が返る
339         """
340
341         with open(self.output_file.upper(), 'r') as f:
342             reader = csv.reader(f, delimiter=" ")
343             data = []
344             for row in reader:
345                 data.append(row)
346
347             volt_tmp = 0.0
348             cnt = 0
349
350             j0_std = 0
351             amplifier_stages = self.stages
352             output_voltage_column = 6
353             squid_start_column = 7
354             jj_per_stages = 4
355
356             pulse_start_time = 0
357             pulse_end_time = 0
358
359             vout_set = []
360             vout_reset = []
361             interval = 1
362             max_interval = 20
363
364             set_start = 0.20
365             set_end = 0.40
366             reset_start = 0.70
367             reset_end = 0.90

```

```

368
369         for row in data:
370             time = float(row[0])
371             if time < 0.4e-9:
372                 j0_std = float(row[squid_start_column])
373                 continue
374
375             if interval > max_interval:
376                 interval -= 1
377                 break
378
379         try:
380             voltage = float(row[output_voltage_column])
381             j0 = float(row[squid_start_column])
382         except Exception as e:
383             info = sys.exc_info() # get error information
384             tbinfo = traceback.format_tb(info[2]) # transfer format
385
386             logger.debug('Handle Exception'.center(40, '='))
387             for tbi in tbinfo:
388                 logger.debug(tbi)
389             logger.debug(' %s' % str(info[1]))
390             logger.debug('time: {}'.format(time))
391             logger.debug('{} {}'.format(start[0], end[0]))
392             logger.debug('{} {}'.format(start[1], end[1]))
393             logger.debug('".center( 40, '=' ))
394             raise
395
396         if math.fabs(j0 - j0_std) >= 2 * math.pi and pulse_start_time == 0:
397             pulse_start_time = time
398             # はの入力間隔 pulse_intervalSFQ => 次のが来るまでは倍の時間がかかる
399             start
400             pulse_end_time = pulse_start_time + self.pulse_interval * 2
401             logger.debug("0% {}, SET_start({})% {}, SET_end({})% {},
402                          RESET_start({})% {}, RESET_end({})% {}, 100% {}".format(
403                             pulse_start_time,
404                             set_start * 1e2,
405                             (pulse_start_time + set_start * (self.pulse_interval * 2)),

```

```

404         set_end * 1e2,
405         (pulse_start_time + set_end * (self.pulse_interval * 2)),
406         reset_start * 1e2,
407         (pulse_start_time + reset_start * (self.pulse_interval * 2)),
408         reset_end * 1e2,
409         (pulse_start_time + reset_end * (self.pulse_interval * 2)),
410         pulse_end_time))
411
412     if pulse_start_time == 0:
413         continue
414
415     # vout_set
416     if time >= (pulse_start_time + set_start * (self.pulse_interval * 2)) and \
417         time <= (pulse_start_time + set_end * (self.pulse_interval * 2)):
418         volt_tmp += voltage
419         cnt += 1
420
421     # after vout_set
422     # で指定した時間を超える vout_set かつの終了時刻との開始時刻の間
423     # vout_setvout_reset
424     if time < (pulse_start_time + (set_end + reset_start)/2 * (self.
425         pulse_interval * 2)) and \
426         time > (pulse_start_time + set_end * (self.pulse_interval * 2)) and cnt >
427         0:
428         vout_set.append(volt_tmp / cnt)
429         logger.debug("{} [{}] {} set: {} ({}).format(time, interval, volt_tmp/cnt, cnt
430             ))
431         cnt = 0
432         volt_tmp = 0
433
434     # vout_reset
435     if time >= (pulse_start_time + reset_start * (self.pulse_interval * 2)) and \
436         time <= (pulse_start_time + reset_end * (self.pulse_interval * 2)) and
437         time <= pulse_end_time:
438         volt_tmp += voltage
439         cnt += 1
440
441     # after vout_reset and new pulse time

```

```

437         if time > pulse_end_time and cnt > 0:
438             vout_reset.append(volt_tmp / cnt)
439             logger.debug("{} {} {} reset: {} ({}).format(time, interval, volt_tmp / cnt,
                    cnt))
440
441             cnt = 0
442             volt_tmp = 0
443             pulse_start_time = time
444             pulse_end_time = pulse_start_time + self.pulse_interval * 2
445             if pulse_end_time > self.sim_time:
446                 break
447             interval += 1
448
449     if len(vout_set) > 0 and len(vout_reset) > 0:
450         vout_set_mean = sum(vout_set)/len(vout_set)
451         vout_reset_mean = sum(vout_reset)/len(vout_reset)
452         vswing = vout_set_mean - vout_reset_mean
453         vout_max = max([x - y for (x, y) in zip(vout_set, vout_reset)])
454         vout_min = min([x - y for (x, y) in zip(vout_set, vout_reset)])
455         vswing_list = [x - y for (x,y) in zip(vout_set, vout_reset)]
456         vswing_mean = sum(vswing_list)/len(vswing_list)
457         vswing_variance = sum([(x - vswing_mean)**2 for x in vswing_list])
458         vswing_stddev = math.sqrt(vswing_variance)
459     else:
460         vout_set_mean = 0
461         vout_reset_mean = 0
462         vswing = 0
463         vout_max = 0
464         vout_min = 0
465         vswing_list = []
466         vswing_mean = 0
467         vswing_variance = 0
468         vswing_stddev = 0
469
470     with open("hoge", 'a') as f:
471         f.write("{} {} {} {} {} {} {} {} {} \n".format(self.now_current, vout_set_mean,
                    vout_reset_mean, vswing, vout_max, vout_min, vswing + vswing_stddev,
                    vswing - vswing_stddev))

```

```

472     logger.debug("{} {} {} {}".format(vout_set_mean, vout_reset_mean, vswing,
        vswing_list))
473     return (vout_set_mean, vout_reset_mean, vout_max, vout_min)
474
475
476 # 結果からのバイアス電流変化時の電圧変化と最大最小電圧値を計算 SQUID
477 def makeVoltage(self):
478     """ 数値計算の結果から Set, 信号入力時の出力電圧を計算 Reset
479     Notes:求め方は、出力電圧を直接見る方法と、位相の時間微分から求める方法のつを出し
480           ている
481           2
482     """
483     pulse_cnt = 0
484     with open(self.output_file.upper(), 'r') as f:
485         reader = csv.reader(f, delimiter=" ")
486         sum = 0
487         cnt = 0
488         data = []
489         for row in reader:
490             data.append(row)
491
492         max = [0.0,0]
493         min = [0.0,0]
494         head = [[],[]]
495         isHead = [True,True]
496         tail = [[],[]]
497         isTail = [True,True]
498         start = [0.0,0.0]
499         end = [0.0,0.0]
500         self.calc_voltage = [0.0,0.0,0.0,0.0]
501         self.calc_voltage2 = [0.0,0.0,0.0,0.0]
502         amplifier_stages = self.stages # アンプの段数
503         squid_startnum = 7 # アンプの記述がのどの列からスタートするか jsim
504         jj_per_stages = 4 # 段あたりの接合数 1
505         for row in data:
506             time = float(row[0])
507             if time < 0.3e-9:
508                 continue

```

```

508
509         try:
510             val = float(row[6])
511
512             if time > (self.pulse_interval*(pulse_cnt+1)):
513                 #pulse_cnt += 1
514                 #tail[3] = head[3] = True
515                 pass
516
517             # 信号 Set立ち上がり ()
518             #start[0] = self.pulse_set + self.pulse_interval*pulse_cnt + self.
519                 tau_set_last + self.tau_squid*amplifier_stages
520             #end[0] = self.pulse_reset + self.pulse_interval*pulse_cnt + self.
521                 tau_set_first + self.tau_squid
522             #start[0] = self.pulse_set + self.pulse_interval * 0.15 + 370e-12 #12
523                 stage
524             #end[0] = self.pulse_reset + 30*1e-12 + self.tau_squid + 200e-12
525             start[0] = self.pulse_set + self.pulse_interval * 0.15 + 260e-12 #4stage
526             end[0] = self.pulse_reset + 30*1e-12 + self.tau_squid + 90e-12
527             #start[0] = self.pulse_set + self.pulse_interval*pulse_cnt + self.jtl_delay
528                 *3
529             #end[0] = self.pulse_reset + self.pulse_interval*pulse_cnt + self.
530                 jtl_delay*3
531             if time >= start[0] and time <= end[0]:
532                 max[1] += 1
533                 max[0] += val
534
535             # 信号が入力されて最後のに入力されたら SetSQUID回のみ (1)
536             if isHead[0] and time >= start[0]:
537                 head[0] = row
538                 isHead[0] = False
539                 logger.debug("[{}]: Input set SFQ reached last stages SQUID".
540                     format(time))
541
542             # 信号が入力されて最初のに入力されたら ResetSQUID
543             if isTail[0] and time >= end[0]:
544                 tail[0] = row
545                 for i in range(amplifier_stages):

```

```

540         for j in range(jj_per_stages):
541             x = squid_startnum + j + jj_per_stages*i
542             a = float(tail[0][x])
543             b = float(head[0][x])
544             c = float(tail[0][0])
545             d = float(head[0][0])
546             if(j<2 and jj_per_stages == 4): # 最初のつの接合は位相変
                化が負なので合わせるためにをかけている 2-1
547                 a *= -1
548                 b *= -1
549             self.calc_voltage[j] += self.kj * ((a - b) / (c - d))
550
551         isTail[0] = False
552         logger.debug("{}: Input reset SFQ reached first stages".format(time
                ))
553
554         # 信号 Reset立ち下がり ()
555         #start[1] = self.pulse_reset + self.pulse_interval*pulse_cnt + self.
                tau_reset_last + self.tau_squid*amplifier_stages
556         #end[1] = self.pulse_set + 2*self.pulse_interval*(pulse_cnt+1) + self.
                tau_reset_first + self.tau_squid
557         #start[1] = self.pulse_reset + self.pulse_interval*pulse_cnt + self.
                pulse_interval * 0.20 + 370e-12 # 12stage
558         #end[1] = self.pulse_set + 2*self.pulse_interval*(pulse_cnt+1) + 200e
                -12
559         start[1] = self.pulse_reset + self.pulse_interval*pulse_cnt + 370e-12 #
                4stage
560         end[1] = self.pulse_set + 2*self.pulse_interval*(pulse_cnt+1) + 200e
                -12
561         #start[1] = self.pulse_reset + self.pulse_interval*pulse_cnt + self.
                jtl_delay*3
562         #end[1] = self.pulse_set + 2*self.pulse_interval*(pulse_cnt+1) + self.
                jtl_delay*3
563         if time >= start[1] and time <= end[1]:
564             #logger.debug("{} {} {}: reset".format(row[0], start, end))
565             min[1] += 1
566             min[0] += val
567

```



```

568         # 信号が入力されて最後のに入力されたら ResetSQUID回のみ (1)
569         if isHead[1] and time >= start[1]:
570             head[1] = row
571             isHead[1] = False
572             logger.debug("{}: Input reset SFQ reached last stages".format(time)
573                          )
574
575         # 信号が入力されて最初のに入力されたら SetSQUID
576         if isTail[1] and time >= end[1]:
577             tail[1] = row
578             for i in range(amplifier_stages):
579                 for j in range(jj_per_stages):
580                     x = squid_startnum + j + jj_per_stages*i
581                     a = float(tail[1][x])
582                     b = float(head[1][x])
583                     c = float(tail[1][0])
584                     d = float(head[1][0])
585                     if(j<2 and jj_per_stages == 4): # 最初のつの接合は位相変
586                                                         化が負なので合わせるためにをかけている 2-1
587                     a *= -1
588                     b *= -1
589                     self.calc_voltage2[j] += self.kj * ((a-b)/(c-d))
590
591             isTail[1] = False
592             logger.debug("{}: Input set SFQ reached first stages".format(time))
593
594     except Exception as e:
595         info = sys.exc_info() # get error infomation
596         tbinfo = traceback.format_tb(info[2]) # transfer format
597
598         logger.debug('Handle Exception'.center(40, '='))
599         for tbi in tbinfo:
600             logger.debug(tbi)
601             logger.debug(' %s' % str(info[1]))
602             logger.debug('time: {}'.format(time))
603             logger.debug('{} {}'.format(start[0], end[0]))
604             logger.debug('{} {}'.format(start[1], end[1]))
605             logger.debug('".center( 40, '=' ))

```

```

604         raise
605
606     self.calc_minmax[0] = max[0]/max[1]
607     self.calc_minmax[1] = min[0]/min[1]
608     #logger.debug("{} {}, {}, {}, {}, {}, {}, {}".format(self.now_current, max[0]/max[1],
609         min[0]/min[1], self.output_file.upper(), self.calc_voltage, self.calc_voltage2,
610         start, end))
611
612     def calcMargin(self):
613         """ バイアス電流を変化させ、その時の出力電圧などを計算、ファイルに出力する
614         """
615         self.parseArgument()
616         with open('margin', 'w') as f:
617             f.write("#Ibias Vout-set Vout-reset Vswing Vswing-max Vswing-min Vout-
618                 set(phase) Vout-reset(phase) Vswing(phase) Vbias freq-input filename\n"
619                 )
620             f.write("#conditions: ./calc_margin.py --input {} --bias_start {} --
621                 bias_end {} --bias_step {} --stages {}\n".format(self.jsim_input, self.
622                 bias_start, self.bias_end, self.bias_step, self.stages))
623         self.now_current = self.bias_start
624         """範囲は
625             bias_start ~ まで bias_end刻み幅はで決まる
626             bias_step
627             gnuplot での見栄えがあまり良くないので最大値から減少させていく方向にした
628         """
629         for i in range((self.bias_start - self.bias_end) / self.bias_step + 1):
630             self.output_file = "trial_{:02d}th.out".format(i)
631             self.modify_file()
632             self.get_pulse_time()
633             self.get_simulate_time()
634             #self.calculate_with_noise(4.2)
635             self.calculate()
636             self.makeIV()
637             logger.debug("[LOG]: {}, {}".format(sum(self.calc_voltage), sum(self.
638                 calc_voltage2)))
639             #self.makeVoltage()
640             (vout_set, vout_reset, vswing_max, vswing_min) = self.makeVoltage_kai()
641             #logger.debug("{} {}".format(self.calc_voltage[0], self.calc_voltage[1]))

```

```

635         buffer = "{} {} {} {} {} {} {} {} {} {} {} {}".format(\
636             self.calc_current, \
637             vout_set, \
638             vout_reset, \
639             vout_set - vout_reset, \
640             vswing_max, \
641             vswing_min, \
642             sum(self.calc_voltage), \
643             sum(self.calc_voltage2), \
644             sum(self.calc_voltage) - sum(self.calc_voltage2), \
645             self.now_current, \
646             self.now_current*1e-2, \
647             self.output_file.upper()
648     )
649     logger.debug(buffer)
650     logger.debug("{}th trial".format(i))
651     f.write(buffer + "\n")
652     self.now_current -= self.bias_step
653     logger.debug("
        =====
        ")

654
655     f.close()
656     shutil.copyfile(self.jsim_temp[1], './usejsm.inp')
657     os.remove(self.jsim_temp[1])
658
659     def main():
660         """ 関数 main
661         """
662         margin = CalcMargin()
663         margin.calcMargin()
664
665         """モジュールからの実行か確認モジュールからだったら何もしない
666
667
668         """
669         if __name__ == '__main__':
670             main()

```
